Basic Studies on Computing Methods in Optimal Control Problems(Dissertation_全文)

AUTHOR(S):

Adachi, Norihiko

BASIC STUDIES ON COMPUTING METHODS IN OPTIMAL CONTROL PROBLEMS

Norimiko ADACHI

1972

BASIC STUDIES ON COMPUTING METHODS IN OPTIMAL CONTROL PROBLEMS

by

Norihiko Adachi

Submitted in partial fulfillment of the requirements for the degree of Doctor of Engineering at Kyoto University.

Kyoto Japan

April 1972

ACKNOWLEDGMENTS

The author would like to express his sincere gratitude to Professor Hidekatsu Tokumaru for his guidance and encouragement in the preparation of this thesis. The author also expresses his thanks to the members of Professor Tokumaru's laboratory.

ABSTRACT

The computational methods in optimization problems are discussed in this thesis.

A generalized variable-metric algorithm is presented. It is shown that this algorithm attains the minimum of a positive definite, quadratic function in a finite number of steps and that the "variable-metric matrix" tends to the inverse of the Hessian matrix. Most known variable-metric methods can be derived from this generalized algorithm, and some new methods are also obtained. This generalized algorithm produces a unique search-direction independently of paraters in the recurrent formulas of variable-metric matrices. And they generate a unique sequence of minimizing points for the given initial conditions if the objective function is quadratic.

These methods are applied to the several simple problems and the computed results are compared with each other.

In the latter part of the thesis an extension of Davidon's method to Hilbert space is presented. The stability and convergence of the method are shown in the case when the functionals to be minimized are quadratic. Similar discussions are made for Fletcher-Reeves's conjugate gradient methods and for one of its variants.

These methods are applied to optimal control problems and two numerical examples are shown. These examples show the superiority of Davidon's method compared with the steepest descent method or the conjugate gradient methods.

CONTENTS

CHAPTER I INTRODUCTION

1.1. Review of Computing Methods in Optimal Control Problems

One of the distinguishable features of the modern control theory is that the concept of optimality plays an important role in it. Unified mathematical approaches for optimal control problems were presented by L. S. Pontryagin (Ref. 1) or by R. Bellman (Ref. 2). Since then, great advances are made in the theory. And now it seems that fundamental properties of optimal control law are clarified, at least so far as deterministic systems are concerned. But it is impossible to obtain optimal control laws or optimal trajectories analytically for the problems of practical interests, except for some particular ones. So that, numerical methods for optimization problems have been studied from the begining of the histry of optimal control. Moreover, more complicated and larger systems are now to be treated in control engineering. And the recent development in large scaled digital computers makes it possible to deal with a large quantity of information in high speed. By these reasons computational method in optimization of control systems becomes of great importance today.

There are two general numerical approaches for the optimal control problems. These are called direct methods and indirect methods. The direct methods improve an initial estimate of the solution sucessively until optimality conditions are satisfied. The indirect methods solve numerically

the equations which are obtained from necessary conditions for optimality. That is, two-points boundary value problems must be solved numerically in indirect methods. Various techniques pertaining to indirect methods are known (Ref. 3-5). In these techniques two-points boundary value problems are transformed to the minimizing problems of functions of finite variables and the transformed problems are solved numerically.

The first direct numerical approach to optimal control problems is an application of the steepest descent method by A. E. Bryson et. al. (Ref. 6) and H. J. Kelly (Ref. 7). Natural extensions of the steepest descent method are the methods which make use of the second order variations of the performance functionals. These methods are called second variation method (Ref. 8-11).

During the last decade, other numerous computational techniques, which belong to direct method, for optimal control problems with various constraints have been presented. A large part of these methods may be regarded as attempts to adapt mathematical programming techniques to optimal control problems. Linear programming is applied to some restricted types of control problems (Ref. 12, 13). Other various well-known nonlinear programming techniques such as Rosen's gradient projection method (Ref. 14) and SUMT of Fiacco and McCormic (Ref. 15) are applied to optimal control problems (Ref. 16-24).

In recent years efficient algorithms for unconstrained minimization problems in the finite dimensional space are

developed. These methods are called conjugate directions methods or variable metric method (Ref. 25-26). By these efficient methods, penalty function methods such as SUMT of Fiacco and McCormic, which transform a given constrained optimization problems to the sequence of unconstrained problems, are becoming practical tools for constrained problems, and said to be most promising method for general constrained problems (Ref. 27). These efficient algorithms in finite dimensional space are extended to the function space and attempted to apply to control problems by several researchers (Ref. 28-30).

1.2. Outline of the Dissertation

Recently remarkable progress in computing methods for unconstrained optimization problems are made. The reasons for this progress are not only an increasing requirement for the solution of large-scale decision problems, but also development of new techniques for constrained optimization problems. These new techniques permit unconstrained optimization procedures to be applied to solve general constrained problems. Numerous new algorithms for unconstrained problems are presented. Most popular ones among them are Fletcher-Reeves's conjugate gradient method and Davidon's method. Most of these new procedures were developed heuristically and relations among them are not clear. In chapter II, a unified approach to unconstrained minimization problems of

functions of several variables are presented. A large part of recently presented algorithms are derived from this general approach. And general properties of these algorithms are discussed. Several particular algorithms are compared with each other by some numerical examples.

In order to apply these algorithms in finite dimensional space to control problems it is necessary to extend these numerical procedures to function spaces. In chapter III Davidon's method and conjugate gradient method, which are most popular among the new algorithms, are extended to a function space. Several important properties of the methods are obtained.

In chapter IV the algorithms discussed in the preceeding chapters are applied to optimal control problems and some numerical examples are shown. Finally some problems concerning the computing method for optimal control problems are discussed.

CHAPTER II

VARIABLE-METRIC METHOD FOR FUNCTION MINIMIZATION PROBLEMS

2.1. Introduction

There are a large number of algorithms for unconstrained optimization problems. One of the oldest and the simplest methods is that of steepest descent, which is said to have been proposed originally by Cauchy in 1874. Another well-known method is Newton's method. Most new algorithms are variants of these two method or are developed on the basis of them. In recent years, a new family of optimization techniques has been proposed, and increasing attentions are being paid to these methods. These methods are called variable-metric (Ref. 31), conjugate-gradient (Ref. 32), or quasi-Newton methods (Ref. 33). Newton's method has an excellent rate of convergence, but the convergence is not always guaranteed, and it requires the second derivatives of the function to be minimized. On the other hand, the steepest-descend method is superior to Newton's method in stability and requires only the first derivatives of the function, but the convergence is often very slow.

From the point of practical computation, the use of second derivatives is undesirable. Therefore, methods which

retain the good characteristics of the previous two methods and use only first derivatives were developed and are being developed. There are two main ideas for this purpose. One is to use conjugacy properties of the search directions, and the another is to approximate the second derivatives of the function by some means. Hence, these methods are called conjugate-gradient methods or quasi-Newton method. Most of the conjugate-gradient methods and quasi-Newton methods are often considered as variable-metric methods from another point of view. Hestens and Stiefel's conjugate-gradient method (Ref. 32) was applied to minimization problems of non-quadratic functions by Fletcher and Reeves (Ref. 25), and projection method (Ref. 34), Davidon's method (Ref. 26, 31) and the rank-one method (Ref. 35) all make use of conjugate properties of the search directions. Davidon's method, proposed by W. C. Davidon and reformulated by R. Fletcher and M. Powell, is one of the most efficient methods and is also called Davidon-Fletcher-Powell (DFP) method (Ref. 35). Several additional algorithms belonging to this family are also available (Ref. 36-40). And unified approaches to this class of algorithms are also attempted (Ref. 38, 39, 41). Since there are a lot of algorithms belonging to this class of algorithms it is important to clarify what is different among these algorithms. From this point of view a result by H. Y. Huang (Ref. 39) is interesting. He has shown that various known algorithms produce a unique sequence of search

directions for a given initial estimate and an initial searching direction if the function to be minimized is quadratic.

In this chapter, a general form of the variable-metric method is presented, and various known algorithms are derived from this general approach. General discussions regarding the convergence of the method for quadratic function are also given.

2.2. Variable-Metric Method

The problem to be considered is to find a local minimum of a function of n variables $x=(x_1, x_2,\ldots, x_n)^T$, where T denotes the transpose of a matrix. The function f(x) is assumed twice differentiable. The gradient of f(x) is denoted by g(x), that is;

$$g(x) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n}\right)^T,$$

and the Hessian matrix of f(x) is denoted by G(x), that is;

$$G(x) = \left(\frac{\partial^2 f}{\partial x_i \partial x_j}\right), \quad i,j=1,2,\ldots, n.$$

The iterative method for this problem is to find a sequence of vectors $(x_0, x_1,\ldots, x_n,\ldots)$ such that

$$x_{i+1} = x_i + \alpha_i p_i \tag{2.1}$$

and

$$\lim_{i\to\infty} g(x_i) = 0. \tag{2.2}$$

The vectors p_i, i=0, 1, 2,... are called search directions. Hereafter x_i, p_i etc. denote vectors at i-th iteration if any comment is not made. In the steepest-descent method,

$$p_i = -g(x_i) \tag{2.3}$$

and in Newton's method,

$$p_i = -G^{-1}(x_i)g(x_i) \ . \tag{2.4}$$

In general, if p_i is defined as

$$p_i = -G_i^{-1}g(x_i)$$

with some matrix G_i then the iterative method is called a variable-metric method, where the n×n matrix G_i is positive definite. A positive-definite matrix G_i defines a metric. Consider the problem of minimizing the directional derivative of f(x) at x_i in the direction p under the constraint that the norm of p is constant, that is,

$$|p|^2 = (p, G_i p) = \text{const.}$$

The solution of this problem is clearly

$$p' = -\alpha G_i^{-1} g(x_i) = \alpha p_i,$$

where α is a suitable scalar. In the steepest-descent method, $G_i = I_n$, where I_n is a unit matrix; and, in

Newton's method, $G_i=G(x_i)$. As in Newton's method, the metric is variable; hence, the iteration method is called the variable-metric method. In (2.1) the i-th step size α_i is determined so that

$$f(x_i + \alpha_i p_i) = \min_{\alpha} f(x_i + \alpha p_i). \tag{2.6}$$

In the following, the vectors s_i and y_i denote $x_{i+1}-x_i$ and $g_{i+1}-g_i$, respectively, that is,

$$s_i = x_{i+1}-x_i, \quad i=0, 1, 2,..., \tag{2.7}$$

$$y_i = g_{i+1}-g_i, \quad i=0, 1, 2,..., \tag{2.8}$$

where g_i denotes $g(x_i)$.

Consider the case of a quadratic function, that is,

$$f(x) = \tfrac{1}{2}(x, Ax) - (g, x), \tag{2.9}$$

where the n×n matrix A is symmetric and (x, y) denotes the inner product of x and y. In this case, by definition,

$$g(x) = Ax - g, \tag{2.10}$$

$$\alpha_i = (p_i, g_i)/(p_i, Ap_i), \tag{2.11}$$

$$y_i = As_i, \tag{2.12}$$

$$\text{or} \quad g_{i+1} = y_i + \alpha_i Ap_i ; \tag{2.13}$$

and, from the choice of α_i, it is clear that

$$(p_i, g_{i+1}) = 0 . \tag{2.14}$$

A set of nonzero vectors $(p_0, p_1,..., p_{n-1})$ is said to be

A-conjugate if

$$(p_i, Ap_j) = 0, \quad i \neq j, \quad i,j=0, 1,\ldots, n-1. \tag{2.15}$$

If A is positive definite, it is clear that n vectors $(p_0, p_1,\ldots, p_{n-1})$ which are A-conjugate are also linearly independent. From (2.13), we have

$$g_n = g_{j+1} + \sum_{i=j+1}^{n-1} \alpha_i Ap_i \; ; \tag{2.16}$$

then, $(g_n, p_j)=0$, $j=0,\ldots, n-1$, using the relations (2.14) and (2.15). Since the vectors p_j, $j=0,\ldots, n-1$, are linearly independent,

$$g_n = 0.$$

In other words, for a positive-definite quadratic form, the minimum is attained at most in n-steps if the search directions are A-conjugate. This is an important property of conjugate vectors; methods using this property are called conjugate-gradient or conjugate-directions methods.

Define n×i matrices Y_i and S_i as follows:

$$\begin{aligned} Y_i &\equiv (y_0, y_i,\ldots, y_{i-1}), \\ S_i &\equiv (s_0, s_1,\ldots, s_{i-1}). \end{aligned} \tag{2.17}$$

Then, the A-conjugate property of the vectors p_i, $(i=0,\ldots, i-1)$, is equivalent to the relations

$$Y_j^T s_j = 0, \quad j=1,\ldots, i-1, \tag{2.18}$$

or $$S_j^T y_j = 0, \quad j=1,\ldots, i-1, \tag{2.19}$$

and, from (2.12),

$$Y_i = AS_i . \tag{2.20}$$

Now, denote the matrix G_i^{-1} in (2.5) by H_i^T; then, the problem is to construct the n×n matrix H_i so that $p_i=-H_i^T g_i$, i=0, 1, 2,..., are a set of A-conjugate vectors. Moreover, it is desirable that H_i tends to A^{-1}. Then, it is expected that in the vicinity of the extremum point, the properties of the algorithm applied to non-quadratic functions are similar to those of Newton's method.

2.3. Projection Algorithm

If $Y_j^T s_j=0$, $j=1, 2,\ldots, i-1$, then the set of vectors s_j, $j=0,\ldots, i-1$, are A-conjugate. Suppose that the vectors s_j, $j=0,\ldots, i-1$ are A-conjugate; then, the i-th variable metric matrix H_i is defined so that vectors $(s_0,\ldots, s_i)$ are A-conjugate;

$$Y_i^T s_i = -\alpha_i Y_i^T H_i^T g_i = 0.$$

A simple idea is to construct H_i such that

$$H_i Y_i = 0, \tag{2.21}$$

that is, $H_i^T g_i$ is orthogonal to the subspace spanned by vectors $(y_0, y_i,\ldots, y_{i-1})$. The algorithm thus derived is

called the projection algorithm (Ref. 34).

Under the assumption that Y_i is of rank i, the algorithm becomes as follows:

$$H_i = R - RY_i(Y_i^T RY_i)^{-1} Y_i^T R, \quad H_0 = R, \tag{2.22}$$

or in recursive form,

$$H_{i+1} = H_i - H_i y_i y_i^T H_i / y_i^T H_i y_i, \quad H_0 = R, \tag{2.23}$$

where R is an arbitrarily given positive-definite, symmetric matrix. For this algorithm, the following result is known (Ref. 36): if the matrix A in (2.9) is positive definite and symmetric, the algorithm (2.22) minimize the quadratic form (2.9) in n or fewer steps.

The Partan method (Ref. 42) and the conjugate-gradient method due to Powell (Ref 43) belong to this class of projection algorithm (Ref. 44).

The general form of algorithm (2.23) will be discussed later.

2.4. Generalized Variable-Metric Algorithm

In the projection algorithm (2.23) the matrices H_i, (i=0, 1,...) tend to zero matrix; $H_n=0$. This property is not what we wanted, as stated in Section 2.1. From (2.20) $A^{-1}Y_i=S_i$; therefore, consider the matrix equation

$$H_i Y_i = S_i, \tag{2.24}$$

where H_i is an unknown (n×n) matrix. If the variable-metric matrix H_i, i=0, 1, 2,..., n, satisfy the above equation at each step, then the A-conjugacy condition of $s_0,\ldots, s_{i-1}$ is the following:

$$Y_j^T s_j = 0, \quad j=1,\ldots, i-1, \tag{2.25}$$

$$\text{or} \quad \alpha_j Y_j^T H_j^T g_j = \alpha_j S_j^T g_j = 0, \quad j=1,\ldots, i-1.$$

To obtain a general solution of Eq.(2.24), the following lemma (Ref. 45) is useful.

Lemma 2.1

A necessary and sufficient condition for the solvability of the matrix equation

$$CXD = E$$

is that $$CC^+ED^+D = E.$$

In this case, the general solution of the equation is

$$X = C^+ED^+ + Y - C^+CYDD^+,$$

where Y is an arbitrary matrix of the same size as X, and where C^+, D^+ are any matrices which satisfy the relations

$$CC^+C = C, \quad DD^+D = D.$$

Applying this lemma, we can write the general solution of Eq.(2.24) as follows:

$$H_i = S_i Y_i^{+} + R_i(I_n - Y_i Y_i^{+}),$$

$$\text{or} \qquad H_i = S_i Y_i^{+} + R_i(I_n - Y_i Y_i^{*}), \tag{2.26}$$

where the matrix R_i is n×n and the matrices Y_i^{+}, Y_i^{*} are i×n. These matrices satisfy the relations $Y_i Y_i^{+} Y_i = Y_i$, $Y_i Y_i^{*} Y_i = Y_i$. The condition that the equation be solvable is that

$$S_i Y_i^{+} Y_i = S_i.$$

Since $S_i = A^{-1} Y_i$, this condition is always satisfied. Next, we shall develop a recurrent formula for H_i in (2.26). Let us define the i×n matrices E_i and the 1×n matrices e_i^T as follows:

$$E_i = -Y_i^{*} y_i c_i (I_n - Y_i Y_i^{*}) / c_i^T (I_n - Y_i Y_i^{*}) y_i, \tag{2.27}$$

$$e_i^T = d_i^T (I_n - Y_i Y_i^{*}) / d_i^T (I_n - Y_i Y_i^{*}) y_i, \tag{2.28}$$

where the vectors c_i and d_i are such that

$$c_i^T (I_n - Y_i Y_i^{*}) y_i \neq 0,$$

$$d_i^T (I_n - Y_i Y_i^{*}) y_i \neq 0.$$

If $Y_i Y_i^{*} = I_n$, we define E_i and e_i as $E_i = (0)$, $e_i^T = (0)$.

Lemma 2.2

The i×n matrices Y_i^{*}, $i \geq 1$, defined recursively by

$$Y_{i+1}^{*} = \begin{pmatrix} Y_i^{*} \\ 0 \end{pmatrix} + \begin{pmatrix} E_i \\ e_i^T \end{pmatrix}, \quad Y_1 = \frac{d_1^T}{d_1^T y_0} \tag{2.29}$$

satisfy the relations

$$Y_i Y_i^{*} Y_i = Y_i, \quad i = 1, 2, \ldots$$

The Lemma 2.2 can be proved by direct calculations.

Suppose that Y_i^+ and Y_i^* in (2.26) are defined by the following recursion formulas:

$$Y_{i+1}^+ = \begin{pmatrix} Y_i^+ + E_{1i} \\ e_{1i}^T \end{pmatrix}, \quad Y_{i+1}^* = \begin{pmatrix} Y_i^* + E_{2i} \\ e_{2i}^T \end{pmatrix}, \tag{2.30}$$

where

$$E_{1i} = -Y_i^+ y_i c_{1i}^T (I_n - Y_i Y_i^+)/c_{1i}^T (I_n - Y_i Y_i^+) y_i,$$

$$e_{1i}^T = d_{1i}^T (I_n - Y_i Y_i^+)/d_{1i}^T (I_n - Y_i Y_i^+) y_i,$$

$$E_{2i} = -Y_i^* y_i c_{2i}^T (I_n - Y_i Y_i^*)/c_{2i}^T (I_n - Y_i Y_i^*) y_i,$$

$$e_{2i}^T = d_{2i}^T (I_n - Y_i Y_i^*)/d_{2i}^T (I_n - Y_i Y_i^*) y_i.$$

At first we consider the case when $R_i = R$ $(i=1, 2, \ldots)$ in the formula (2.26). Then, the following recurrent formula for H_i in (2.26) is obtained:

$$H_{i+1} = H_i + \frac{s_i d_{1i}^T (I_n - Y_i Y_i^+)}{d_{1i}^T (I_n - Y_i Y_i^+) y_i} - \frac{S_i Y_i^+ y_i c_{1i}^T (I_n - Y_i Y_i^+)}{c_{1i}^T (I_n - Y_i Y_i^+) y_i}$$

$$-R\left\{\frac{y_i d_{2i}^T (I_n - Y_i Y_i^*)}{d_{2i}^T (I_n - Y_i Y_i^*) y_i} - \frac{Y_i Y_i^* y_i c_{2i}^T (I_n - Y_i Y_i^*)}{c_{2i}^T (I_n - Y_i Y_i^*) y_i}\right\}. \tag{2.32}$$

Let us denote the matrices $I_n - Y_i Y_i^+$ and $I_n - Y_i Y_i^*$ by J_i and K_i, respectively. Then, the recursion formula (2.32) can be written as,

$$H_{i+1} = H_i + \frac{s_i d_{1i}^T J_i}{d_{1i}^T J_i y_i} - \frac{(H_i - RK_i) y_i c_{1i}^T J_i}{c_{1i}^T J_i y_i}$$

$$-R\{\frac{y_i d_{2i}^T K_i}{d_{2i}^T K_i y_i} - \frac{(I_n - K_i) y_i c_{2i}^T K_i}{c_{2i}^T K_i y_i}\}, \quad H_0 = R, \tag{2.33}$$

where

$$J_{i+1} = J_i - \{\frac{(J_i - I_n) y_i c_{1i}^T J_i}{c_{1i}^T J_i y_i} + \frac{y_i d_{1i}^T J_i}{d_{1i}^T J_i y_i}\}, \quad J_0 = I_n, \tag{2.34}$$

and

$$K_{i+1} = K_i - \{\frac{(K_i - I_n) y_i c_{2i}^T K_i}{c_{2i}^T K_i y_i} + \frac{y_i d_{2i}^T K_i}{d_{2i}^T K_i y_i}\}, \quad K_0 = I_n. \tag{2.35}$$

The algorithm with matrices H_i, i=0, 1, 2,..., defined by the formulas (2.33)–(2.35) is a most general form of the variable-metric method.

If $c_{1i}=d_{1i}=c_i$ and $c_{2i}=d_{2i}=d_i$, i=1, 2,..., in (2.33)–(2.35), then

$$H_{i+1} = H_i + \frac{(s_i - H_i y_i + RK_i y_i) c_i^T J_i}{c_i^T J_i y_i} - \frac{RK_i y_i d_i^T K_i}{d_i^T K_i y_i}, \quad H_0 = R, \tag{2.36}$$

where

$$J_{i+1} = J_i - \frac{J_i y_i c_i^T J_i}{c_i^T J_i y_i}, \quad J_0 = I_n,$$

$$K_{i+1} = K_i - \frac{K_i y_i d_i^T K_i}{d_i^T K_i y_i}, \quad K_0 = I_n.$$

If $J_i=K_i$ in (2.33), that is $c_{1i}=c_{2i}=c_i$, $d_{1i}=d_{2i}=d_i$, i=1, 2,..., then

$$H_{i+1} = H_i = \frac{(s_i-Ry_i)d_i^T J_i}{d_i^T J_i y_i} - \frac{(H_i-R)y_i c_i^T J_i}{c_i^T J_i y_i}, \quad H_0=R,$$
$$J_{i+1} = J_i - \{\frac{(J_i-I_n)y_i c_i^T J_i}{c_i^T J_i y_i} + \frac{y_i d_i^T J_i}{d_i^T J_i y_i}\}. \tag{2.37}$$

If $c_i=d_i$ in (2.36) or (2.37), then

$$H_{i+1} = H_i + \frac{(s_i-H_i y_i)c_i^T J_i}{c_i^T J_i y_i}, \quad H_0=R,$$
$$J_{i+1} = J_i - \frac{J_i y_i c_i^T J_i}{c_i^T J_i y_i}, \tag{2.38}$$

where the vectors c_i, i=1, 2,..., are chosen so that $c_i^T J_i y_i \neq 0$. It is natural to choose c_i from an orthonormal basis $\{e_j\}$ of R^n; then,

$$H_{i+1} = H_i + \frac{(s_i-H_i y_i)e_{ji}^T J_i}{e_{ji}^T J_i y_i},$$
$$J_{i+1} = J_i - \frac{J_i y_i e_{ji}^T J_i}{e_{ji}^T J_i y_i}, \tag{2.39}$$

where e_{ji} is chosen from the set $\{e_j: j=1, 2,..., n\}$ so that $e_{ji}^T J_i y_i \neq 0$ at the i-th step. This algorithm is similar to that of Murtagh and Sargent (Ref. 46), but there is some difference in the recursion formula for J_i. In (2.38), if a vector y_i is taken as c_i at the i-th step, clearly the

recursion formula for J_i is identical with that of the projection algorithm stated in section 2.2. Therefore, if the symbols H_i and J_i in (2.38) are interchanged and if c_i is set as y_i, another algorithm, known as the projected Newton's method (Ref. 36), is obtained:

$$H_{i+1} = H_i - \frac{H_i y_i y_i^T H_i}{y_i^T H_i y_i} , \tag{2.40}$$

$$J_{i+1} = J_i - \frac{(s_i - J_i y_i) y_i^T H_i}{y_i^T H_i y_i} ;$$

when i is a multiple of n, H_i is reset as J_i, that is, $H_i = J_i$; i=jn, j=0, 1,... .

Now, we state a generalization of Projection Algorithm (2.23). From the preceeeding discussions a general solution of the equation (2.21) considered H_i as a unknown matrix is written as

$$H_i = R_i (I_n - Y_i Y_i^*) . \tag{2.41}$$

Following the same procedure as stated above, the following resursion formula for H_i in (2.41) is obtained,

$$H_{i+1} = H_i - R \left\{ \frac{y_i d_{2i}^T K_i}{d_{2i}^T K_i y_i} - \frac{(I_n - K_i) y_i c_{2i}^T k_i}{c_{2i}^T K_i y_i} \right\} , \quad H_0 = R, \tag{2.42}$$

$$\text{or} \quad H_{i+1} = H_i - \left\{ \frac{(H_i - R) y_i c_i^T H_i}{c_i^T H_i y_i} + \frac{R y_i d_i^T H_i}{d_i^T H_i y_i} \right\} ,$$

where $R^{-1} c_{2i}$ and $R^{-1} d_{2i}$ are denoted by c_i and d_i respectively for simplicity and chosen so that

$$c_i^T H_i y_i \neq 0, \quad d_i^T H_i y_i \neq 0.$$

This algorithm is a generalization of Algorithm (2.23). Clearly (2.23) is obtained if y_i is substituted for c_i and d_i in (2.42).

2.5. Greenstadt's Method

Unified approaches to the variable-metric method are presented by few authors. H. Y. Huang solved essentially the matrix equation with a parameter ρ, that is,

$$H_i Y_i = \rho S_i,$$

in a restricted form. Various particular algorithms are obtained from this approach (Ref. 39). Another general approach to variable-metric algorithm is presented by J. Greenstadt (Ref. 37).

In this section, we consider Greenstadt's algorithm from the point of the preceeding discussions. He solved the problem of minimizing the norm of a matrix E,

$$N(E) \equiv T_r(WEWE^T),$$

subject to the constraints

$$E^T = E, \quad (E+H_i)y_i = s_i,$$

where $E=H_{i+1}-H_i$. Here, W is a positive-definite, symmetric matrix and T_r means the trace of a matrix. He derived the following formula:

$$H_{i+1} = H_i + \frac{1}{y_i^T M y_i}\{s_i y_i^T M + M y_i s_i^T - H_i y_i y_i^T M$$

$$- M y_i y_i^T H_i - \frac{1}{y_i^T M y_i}(y_i^T s_i - y_i^T H_i y_i) M y_i y_i^T M\}, \qquad (2.43)$$

where $M=W^{-1}$.

By special choices of the matrix M, several variable-metric algorithm can be derived (Ref. 37-38). Here, we shall show that Greenstadt's algorithm (2.43) can be obtained from the same approach as in the preceeding section. In Section 2.3, the matrix equation (2.24) is solved and the general solution is expressed as in (2.26):

$$H_i = S_i Y_i^* + R_i(I_n - Y_i Y_i^*),$$

where the i×n matrix Y_i^* is such that

$$Y_i Y_i^* Y_i = Y_i.$$

Now, rewrite the above formula in the following form:

$$H_i = S_i Y_i^* + (S_i Y_i^*)^T - (S_i Y_i^*)^T + R_i(I_n - Y_i Y_i^*),$$

and choose R_i so that the corresponding variable-metric matrix H_i becomes symmetric. A simple choice of R_i is to take $(S_i Y_i^*)^T$ as R_i. Then

$$H_i = (S_i Y_i^*) + (S_i Y_i^*)^T - (S_i Y_i^*)^T (Y_i Y_i^*). \qquad (2.44)$$

Since

$$(S_i Y_i^*)^T (Y_i Y_i^*) = Y_i^{*T} S_i^T Y_i Y_i^* = Y_i^{*T} Y_i^T A^{-1} Y_i Y_i^*,$$

matrix H_i becomes symmetric.

Using the recursion formula for Y_i^* in Lemma 2.2 with $c_i=d_i$ and assuming that the vectors $s_0,\ldots, s_i$ are A-conjugate to each other, we see that H_i in (2.43) is rewritten recursively as

$$H_{i+1} = H_i + \frac{(s_i-H_iy_i)c_i^TJ_i}{c_i^TJ_iy_i} + \frac{J_i^Tc_i(s_i-H_iy_i)^T}{c_i^TJ_iy_i}$$

$$- \frac{J_i^Tc_i(s_i^Ty_i-y_i^TH_iy_i)c_i^TJ_i}{(c_i^TJ_iy_i)^2}, \qquad (2.45)$$

where $J_i=I_n-Y_iY_i^*$ and where c_i is any vector such that $c_i^TJ_iy_i\neq 0$. This formula is analogous to that in Greenstadt's algorithm; his first particular formula is obtained by substituting $H_i^Ty_i$ in place of c_i in (2.45) (Ref. 37).

We note that Greenstadt's general algorithm can be obtained directly from another approach. In the above discussion, we solved the equation with unknown matrix H_i,

$$H_iY_i = S_i .$$

Here, we shall solve the equation

$$H_{i+1}y_i = s_i , \qquad (2.46)$$

where the unknown is the matrix H_{i+1}. This equation is equivalent to

$$(H_{i+1}-H_i)y_i = s_i-H_iy_i.$$

Then, by Lemma 2.1,

$$H_{i+1}-H_i = (s_i-H_iy_i)y_i^*+R_i(I_n-Y_iy_i^*), \tag{2.47}$$

where $1\times n$ matrix y_i^* is such that

$$y_iy_i^*y_i = y_i.$$

In general, y_i^* is of the form

$$y_i^* = y_i^TM/y_i^TMy_i,$$

where the matrix M is $n\times n$. Substituting this expression of y_i^* in (2.47) yields

$$H_{i+1} = H_i+(s_i-H_iy_i)\frac{y_i^TM}{y_i^TMy_i} + R_i(I_n-\frac{y_iy_i^TM}{y_i^TMy_i}).$$

If R_i is defined as

$$R_i = \frac{My_i(s_i-H_iy_i)^T}{y_i^TMy_i},$$

then

$$H_{i+1} = H_i + \frac{1}{y_i^TMy_i}\{(s_i-H_iy_i)y_i^TM+My_i(s_i-H_iy_i)^T$$

$$-\frac{1}{y_i^TMy_i}My_i(s_i^Ty_i-y_i^TH_iy_i)y_i^TM\}. \tag{2.48}$$

This formula is identical with that of Greenstadt.

In closing, we note that, if H_i is defined by (2.45), the relation $H_iy_j=s_j$, $j=0, 1, 2,\ldots, i-1$, are ensured; but if H_i is defined by (2.48), only the relation $H_iy_{i-1}=s_{i-1}$ is valid in general.

2.6. One-Parameter Family of Variable- Metric Method

Let n×n matrix H_i be the i-th variable-metric matrix such that $H_iY_i=S_i$. Then, a necessary and sufficient condition for $H_{i+1}=H_i+\Delta H_i$ to be an (i+1)-th variable-metric matrix of the same property is that $\Delta H_iY_i=0$ and $\Delta H_iy_i=s_i-H_iy_i$. Suppose that H_i is an i-th variable matrix and that $H_{i+1}^{1} = H_i+\Delta H_i^{1}$ and $H_{i+1}^{0} = H_i+\Delta H_i^{0}$ are (i+1)-th variable-metric matrices. Then, it is clear that

$$H_{i+1}^{\lambda} = \lambda H_{i+1}^{1}+(1-\lambda)H_{i+1}^{0}$$

$$= H_i+\lambda\Delta H_i^{1}+(1-\lambda)\Delta H_i^{0}$$

is also a variable-metric matrix at the (i+1)-th step such that $H_{i+1}^{\lambda}Y_{i+1} = S_{i+1}$, where λ is any scalar. From this relation, a family of variable-metric algorithms, which depend on one parameter λ, is generated.

The first one-parameter family of variable-metric algorithms was obtained by C. G. Broyden (Ref. 33); D. Goldfarb has shown that all known symmetric, variable-metric matrices are generated by those of the DFP method (Ref. 35) and rank-one method (Ref. 33) for special choices of a parameter (Ref. 38).

By the same procedure a one-parameter family of algorithms can be generated from two variable-metric algorithms in which relation $H_iY_i=0$ (i=1, 2,...) are satisfied.

2.7. General Convergence Properties

In the preceding sections, several general variable-metric algorithms were presented. Here, we shall show some properties of those methods. Let us define the most general variable-metric algorithm as follows:

$$x_{i+1} = x_i + s_i \tag{2.49}$$

$$s_i = \alpha_i p_i \tag{2.50}$$

$$p_i = -H_i^T p_i \tag{2.51}$$

$$H_i = \mu S_i Y_i^+ + R_i(I_n - Y_i Y_i^*), \quad H_0 = R, \tag{2.52}$$

where $Y_i Y_i^+ Y_i = Y_i$, $Y_i Y_i^* Y_i = Y_i$.

Here R, R_i, $i=1, 2,\ldots$, are $n \times n$ matrices and μ is a scalar parameter, and α_i is a scalar such that

$$f(x_i + \alpha_i p_i) = \min_{\alpha} f(x_i + \alpha p_i). \tag{2.53}$$

Clearly, this formulation includes all of the algorithms presented in Section 2.3 and 2.4. Therefore, we shall present properties for the algorithm defined by (2.49)-(2.52), applied to the quadratic form (2.9).

Theorem 2.1

The following relations hold:

(i) $S_i^T g_i = 0$, $i=1, 2,\ldots$;

(ii) vectors $s_0, s_1, \ldots$, are A-conjugate, that is,

$$s_i^T A s_j = 0 \quad \text{if} \quad i \neq j.$$

proof

At first, we shall note that $H_i Y_i = \mu S_i$ $(i=1, 2, \ldots)$ in the algorithm (2.49)-(2.52). It is clear that $s_0^T g_1 = 0$, because of the choice of the step-size α_0; also,

$$S_1^T g_1 = 0.$$

Let us assume that $S_i^T g_i = 0$. Then, $S_i^T g_{i+1} = 0$, since

$$S_i^T g_{i+1} = S_i^T (g_i + y_i) = S_i^T y_i = (AS_i)^T s_i = -\mu\alpha_i S_i^T g_i = 0.$$

Hence

$$S_{i+1}^T g_{i+1} = \begin{pmatrix} S_i^T g_{i+1} \\ s_i^T g_{i+1} \end{pmatrix} = 0.$$

This proves the first part of the theorem. Then it is sufficient to prove that $Y_i^T s_i = 0$, $i=1, 2, \ldots$. And this is clear from the first part of the theorem, since

$$Y_i^T s_i = -\mu\alpha_i S_i^T g_i .$$

If $\mu=0$ we see, from the above proof that the conjugate property of vectors $s_0, s_1, \ldots$, hold even if α_i $(i=0, 1, \ldots)$ are taken arbitrarily. This is an important property of the algorithms derived from the relation; $H_i Y_i = 0$.

Theorem 2.2

If (1) A is a positive-definite matrix and (2) $s_i \neq 0$ for non-zero g_i, then $g_j=0$ $(j \leq n-1)$ or $g_n=0$ and $H_n=\mu A^{-1}$; that is, the minimum of a positive-definite quadratic function is attained at most with n iterations.

proof

Suppose that $s_0, s_1, \ldots, s_{n-1}$ are all non-zero vectors. Since A is positive definite $s_0, s_1, \ldots, s_{n-1}$ are linearly independent. Hence, $g_n=0$, since $S_n^T g_n=0$ from Theorem 2.1. Moreover, $H_n Y_n=\mu S_n$, and the matrix Y_n is of rank n, so that $H_n=\mu A^{-1}$. This completes the theorem.

From Theorem 2.2 if s_n does not vanish at non-extremum point, the convergence of the general iteration scheme (2.49)—(2.52) is ensured for positive definite quadratic forms. But the above condition dose not seem to hold in general. We shall discuss the condition later again.

2.8. Particular Variable-Metric Algorithms

In this Section, various algorithms, most of which are known already, are derived from the generalized variable-metric algorithms presented in the preceding sections. In the Section 2.3-2.4 we have obtained three fundamental algorithms, which are denoted by Algorithm (A), Algorithm (B) and Algorithm (C):

Algorithm (A)

$$H_{i+1} = H_i + \frac{s_i d_{1i}^T J_i}{d_{1i}^T J_i y_i} - \frac{(H_i - R_i K_i) y_i c_{1i}^T J_i}{c_{1i}^T J_i y_i}$$

$$-R\{\frac{y_i d_{2i}^T K_i}{d_{2i}^T K_i y_i} - \frac{(I_n - K_i) y_i c_{2i}^T K_i}{c_{2i}^T K_i y_i}\}, \quad H_0 = R, \qquad (2.54)$$

$$J_{i+1} = J_i - \{\frac{(J_i - I_n) y_i c_{1i}^T J_i}{c_{1i}^T J_i y_i} + \frac{y_i d_{1i}^T J_i}{d_{1i}^T J_i y_i}\}, \quad J_0 = I_n \qquad (2.55)$$

$$K_{i+1} = K_i - \{\frac{(K_i - I_n) y_i c_{2i}^T K_i}{c_{2i}^T K_i y_i} + \frac{y_i d_{2i}^T K_i}{d_{2i}^T K_i y_i}\}, \quad K_0 = I_n, \qquad (2.56)$$

where c_{ji} $(j=1, 2,)$ and d_{ji} $(j=1, 2,)$ are any n-dimensional vectors such that $c_{1i}^T J_i y_i \neq 0$, $c_{2i}^T J_i y_i \neq 0$, $d_{1i}^T J_i y_i \neq 0$ and $d_{2i}^T K_i y_i \neq 0$ and I_n is an $(n \times n)$ unit matrix.

Algorithm (B)

$$H_{i+1} = H_i + \frac{(s_i - H_i y_i) c_i^T J_i}{c_i^T J_i y_i} + \frac{J_i^T c_i (s_i - H_i y_i)^T}{c_i^T J_i y_i}$$

$$- \frac{J_i^T c_i (s_i^T y_i - y_i^T H_i y_i) c_i^T J_i}{(c_i^T J_i y_i)^2}, \quad H_0 = R. \qquad (2.57)$$

$$J_{i+1} = J_i - \frac{J_i y_i c_i^T J_i}{c_i^T J_i y_i}, \quad J_0 = I_n, \qquad (2.58)$$

where c_i is any vector such that

$$c_i^T J_i y_i \neq 0$$

Algorithm (C)

$$H_{i+1} = H_i - \frac{(H_i - R)y_i c_i^T H_i}{c_i^T H_i y_i} - \frac{R y_i d_i^T H_i}{d_i^T H_i y_i}, \quad H_0 = R, \tag{2.59}$$

where c_i and d_i are chosen so that $c_i^T H_i y_i \neq 0$ and $d_i^T H_i y_i \neq 0$. In the above algorithms R is an arbitrary (n×n) matrix and assumed to be symmetric.

Various algorithms are derived from the above generalized algorithms by particular choices of parameters:

Algorithm I (Pearson)

Substituting $H_i^T y_i$ for c_i in (2.38), or $H_i^T y_i$ for c_{1i}, c_{2i}, d_{1i} and d_{2i} in (2.55)-(2.57), leads to

$$H_{i+1} = H_i + \frac{(s_i - H_i y_i) y_i^T H_i J_i}{y_i^T H_i J_i y_i}, \quad H_0 = R.$$

Since

$$y_i^T H_i J_i y_i = y_i^T H_i y_i,$$

we have an algorithm,

$$H_{i+1} = H_i + \frac{(s_i - H_i y_i) y_i^T H_i}{y_i^T H_i y_i} \tag{2.60}$$

Algorithm II (McCormic)

Set $s_i = c_i$ in (2.38). Since $c_i^T J_i = s_i^T (I_n - Y_i Y_i^*) = s_i^T$

$$H_{i+1} = H_i + \frac{(s_i - H_i y_i) s_i^T}{s_i^T y_i}. \tag{2.61}$$

Algorithm III (Rank-One Method)

Set $c_i=H_i^T y_i-s_i$ in (2.38). In this case,

$$c_i^T J_i = -(s_i-H_i^T y_i)^T(I_n-Y_i Y_i^*) = -(s_i-H_i^T y_i)^T.$$

Hence,

$$H_{i+1} = H_i + \frac{(s_i-H_i y_i)(s_i-H_i^T y_i)^T}{(s_i-H_i y_i)^T y_i}.$$

If R is symmetric, the matrices H_i, $i\geq 1$, are symmetric, so that

$$H_{i+1} = H_i + \frac{(s_i-H_i y_i)(s_i-H_i y_i)^T}{(s_i-H_i y_i)^T y_i}. \qquad (2.62)$$

This method is known as the rank-one method.

Algorithm IV (DFP method)

Consider the more general algorithm define by (2.36) which is derived from Algorithm (A) by setting $c_{1i}=d_{1i}=c_i$ and $c_{2i}=d_{2i}=d_i$. Replace s_i and $H_i^T y_i$ for c_i and d_i in the formula (2.36), respectively. Then, a particular algorithm is derived. In this case, it is easily proved that the matrices $S_i Y_i^*$, $i\geq 1$, are symmetric. Using this property, we obtain

$$H_{i+1} = H_i + \frac{s_i s_i^T}{s_i^T y_i} - \frac{H_i y_i y_i^T H_i}{y_i^T H_i y_i}. \qquad (2.63)$$

This is well-known Davidon-Fletcher-Powell algorithm.

Algorithm V

Let us substitute s_i and $H_i^T y_i$ for c_i and d_i in the formula (2.37), respectively; then,

$$H_{i+1} = H_i + \frac{(s_i - Ry_i)y_i^T H_i}{y_i^T H_i y_i} - \frac{(H_i - R)y_i s_i^T}{s_i^T y_i} . \qquad (2.64)$$

Algorithm VI

Set $c_i = H_i^T y_i$, $d_i = s_i$ in the formula (2.37), that is, $c_{1i} = c_{2i} = H_i^T y_i$ and $d_{1i} = d_{2i} = s_i$ in Algorithm (A). In this case, we obtain the following algorithm:

$$H_{i+1} = H_i + \frac{(s_i - Ry_i)s_i^T}{s_i^T y_i} - \frac{(H_i - R)y_i y_i^T H_i}{y_i^T H_i y_i} . \qquad (2.65)$$

Algorithm VII

In Algorithm V and VI, an initial matrix R is present, that is not desirable. From these two algorithms, another algorithm which does not contain the matrix R can be derived. From the discussion in Section 2.5, we see that

$$H_{i+1} = H_i + \lambda\{\frac{(s_i - Ry_i)s_i^T}{s_i^T y_i} - \frac{(H_i - R)y_i y_i^T H_i}{y_i^T H_i y_i}\}$$

$$+ (1-\lambda)\{\frac{(s_i - Ry_i)y_i^T H_i}{y_i^T H_i y_i} - \frac{(H_i - R)y_i s_i^T}{s_i^T y_i}\}$$

is a one-parameter family of variable-metric algorithms. If $\lambda = 1/2$, we obtain the following algorithm:

$$H_{i+1} = H_i + \frac{1}{2}\left\{\frac{(s_i - H_i y_i)s_i^T}{s_i^T y_i} + \frac{(s_i - H_i y_i)y_i^T H_i}{y_i^T H_i y_i}\right\}. \quad (2.66)$$

This result can be obtained from Algorithm I and Algorithm II by the same approach.

Algorithm VIII (Greenstadt)

If $H_i^T y_i$ is substituted for c_i in Algorithm (B), we obtain Greenstadt's first algorithm, that is,

$$H_{i+1} = H_i + \frac{1}{y_i^T H_i y_i}\{s_i y_i^T H_i + H_i y_i s_i^T - (1 + \frac{y_i^T s_i}{y_i^T H_i y_i}) H_i y_i y_i^T H_i\} . \quad (2.67)$$

Algorithm IX (Goldfarb)

If $c_i = s_i$ in Algorithm (B) we have

$$H_{i+1} = H_i + \frac{1}{y_i^T s_i}\{-s_i y_i^T H_i - H_i y_i s_i^T + (1 + \frac{y_i^T H_i y_i}{y_i^T s_i}) s_i s_i^T\}. \quad (2.68)$$

This slgorithm was obtained by GoldFarb (Ref. 38).

Algorithm X (Projection Method)

If $c_i = d_i = R^{-1} H_i^T y_i$ in Algorithm (C),

$$c_i^T H_i = y_i^T H_i R^{-1} H_i = y_i^T H_i .$$

Then

$$H_{i+1} = H_i - \frac{H_i y_i y_i^T H_i}{y_i^T H_i y_i} . \tag{2.69}$$

Algorithm XI (Huang)

If $c_i=d_i=R^{-1}s_i$ in Algorithm (C)

$$H_{i+1} = H_i - \frac{H_i y_i s_i^T}{s_i^T y_i} . \tag{2.70}$$

Algorithm XII (Huang)

If $c_i=d_i=R^{-1}(s_i-H_i^T y_i)$ in Algorithm (C)

$$H_{i+1} = H_i - \frac{H_i y_i (s_i-H_i^T y_i)^T}{(s_i-H_i^T y_i)^T y_i} . \tag{2.71}$$

In the above, Algorithm I-VII are pertaining to the first class of the generalized algorithms and Algorithm VIII and IX to the class of Algorithm (B) and Algorithm X-XII to the class of Algorithm (C)

2.9. Uniqueness of Search Direction

2.9.1

In this section we shall show that all of the algorithms in the preceeding section produce the same search directions at each steps for the same initial point x_0 and initial matrix $H_0=R$. Discussions are restricted to the quadratic function (2.9).

Theorem 2.3

Let's α_{ji}, β_{ji} (i=1, 2, 3), α_i and β_i be scalor parameters. If

$$c_{ji} = \alpha_{ji} H_i^T y_i + \beta_{ji} s_i, \quad (j=1, 2)$$
$$d_{2i} = \alpha_{3i} H_i^T y_i + \beta_{3i} s_i \tag{2.72}$$

in Algorithm (A),

$$c_i = \alpha_i H_i^T y_i + \beta_i s_i$$
and $$H_i^T = H_i \tag{2.73}$$

in Algorithm (B) and if

$$c_i = R^{1}(\alpha_{1i} H_i^T y_i + \beta_{1i} s_i)$$
$$d_i = R^{-1}(\alpha_{2i} H_i^T y_i + \beta_{2i} s_i) \tag{2.74}$$

in Algorithm (C), then for all of these slgorithms

$$p_{i+1} = -H_{i+1}^T g_{i+1} = \gamma_{i+1} q_{i+1}, \tag{2.75}$$

$$q_{i+1} \equiv (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}, \tag{2.76}$$

where γ_{i+1} is determined for Algorithm (A), (B) and (C) respectively, and depends on parameters in (2.72)-(2.74).

proof (see Appendix of this chapter)

From this theorem if a symmetric matrix H_i is given, then (i+1)-th search direction generated by Algorithm (A), (B) and (C) is uniquely determined independently of the Algorithms and parameters included.

Theorem 2.4

If vectors $(s_0, s_1, \ldots, s_i)$ are defined and all of them are non-zero vectors, then

$$q_{i+1} \equiv (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}$$

$$= (I - \sum_{j=0}^{i} \frac{s_j y_j^T}{s_j^T y_j}) R^T g_{i+1} \quad , \qquad (2.77)$$

under the same choices of parameters as in Theorem 1.

proof

At first it is noted that $H_i^T y_i$ is a linear combination of vectors s_i and s_{i+1}, since

$$H_i^T y_i = H_i^T g_{i+1} - H_i^T g_i$$

$$= H_i^T g_{i+1} - \frac{s_i^T y_i}{s_i^T y_i} H_i^T g_i$$

$$= (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1} + \frac{y_i^T H_i y_i}{s_i^T y_i} s_i$$

$$= \alpha s_{i+1} + \frac{y_i^T H_i y_i}{s_i^T y_i} s_i ,$$

where α is a suitable scalar. Then for Algorithm (A) (C) with parameters defined by (2.72) (2.74),

$$q_{i+1} \equiv (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}$$

$$= H_{i-1}^T g_{i+1} + \eta_i s_i + \eta_{i-1} s_{i\ 1} ,$$

if we take appropriate scalars η_i and $\eta_{i\ 1}$. Since $(s_0, s_1, \ldots, s_i)$ are conjugate

$$y_i^T q_{i+1} = y_{i-1}^T q_{i+1} = 0.$$

Therefore $$y_i^T H_{i-1}^T g_{i+1} + \eta_i s_i^T y_i = 0,$$

$$y_{i-1}^T H_{i-1}^T g_{i+1} + \eta_{i-1} s_{i-1}^T y_{i\ 1} = 0,$$

that is,

$$\eta_i = - \frac{y_i^T H_{i-1}^T g_{i+1}}{s_i^T y_i} , \quad \eta_{i\ 1} = - \frac{y_{i-1}^T H_{i-1}^T g_{i+1}}{s_{i-1}^T y_{i-1}} .$$

Hence,

$$q_{i+1} = (I_n - \frac{s_{i-1} y_{i-1}^T}{s_{i-1}^T y_{i-1}} - \frac{s_i y_i^T}{s_i^T y_i}) H_{i-1}^T g_{i+1}.$$

Repeating the same procedures,

$$q_{i+1} = (I_n - \sum_{j=0}^{i} \frac{s_j y_j^T}{s_j^T y_j}) R^T g_{i+1}.$$

Q.E.D.

Clearly we have the following result from the above theorem.

Corollary

Under the same condition of Theorem 2

$$q_i^T g_i = g_i^T H_{i-1}^T g_i = \cdots = g_i^T R g_i \ .$$

It is clear from Theorem 2.4 that for a given initial point x_0 and initial matrix R all particular algorithms, derived from Algorithm (A), (B) and (C) by choosing parameters as in Theorem 2.3, generate a unique sequence of the search directions $(q_0, q_1, \ldots)$ and the corresponding unique sequence of minimizing points $(x_0, x_1, \ldots)$. If the initial matrix R is positive or negative definite, the above Corollary ensures that the algorithms are stable for quadratic functions, that is;

$$f(x_{i+1}) \leq f(x_i), \quad (i=0, 1, \ldots) \ .$$

Theorem 2.3 and Theorem 2.4 are generalizations of results obtained by H. Y. Huang (Ref. 39).

Suppose that H_i is an i-th variable-metric matrix and that $H_{i+1}^0 = H_i + E_i^0$ and $H_{i+1}^1 = H_i + E_i^1$ are (i+1)-th variable metric matrices. Then, from the discussions of Section 2.5, a family of variable-metric algorithms, which depends on one parameter λ, is generated;

$$H_{i+1}^{\lambda} = H_i^{\lambda} + \lambda E_i^1 + (1-\lambda) E_i^0$$

$$H_0^{\lambda} = R.$$

Theorem 2.5

If the minimizing algorithms with variable-metric matrix H_i^0 and H_i^1 have the properties represented by the formulas (2.75)-(2.77), then the sequence of the search directions and the corresponding sequence of the minimizing points generated by the algorithms with H_i^λ does not depend on the parameter λ.

proof

From Theorem 2.4,

$$p_{i+1} = -H_{i+1}^{T\,\lambda} g_{i+1}$$

$$= -\lambda H_{i+1}^{T\,1} g_{i+1} - (1-\lambda) H_{i+1}^{T\,0} g_{i+1}$$

$$(\lambda\gamma_{i+1}^{1} + (1-\lambda)\gamma_{i+1}^{0})(I_n - \sum_{\gamma=0}^{i} \frac{s_\gamma y_\gamma^T}{s_\gamma^T y_\gamma}) R^T g_{i+1},$$

where γ_{i+1}^{1} and γ_{i+1}^{0} are suitable scalars. Hence, searching directions does not depend on λ and the sequence of search directions depend only on an initial estimate x_0 and an initial matrix R. Cosequently, the sequence of minimizing points is also independent of λ.

Theorem 2.6

Suppose that the extremum point x^* of the given quadratic function (2.9) is obtained by the Algorithm I-XII after n-times iterations for a given initial point x_0 and an initial matrix R. Then, the generated sequences of the searching point (x_0, x_1,..., x_{n-1}, x^*) is the same for all the algorithms.

proof

Let's substitute the parameters given in the formulas (2.72)-(2.73) for Algorithm (A)-(C) respectively. Then it can be easily proved that Algorithm I-VI and VIII-XII are derived from the general algorithm by particular choices of parameters; α_{ji}, β_{ji} (j=1, 2, 3), α_i and β_i. Algorithm VII is obtained from the one parameter family of variable metric algorithms generated from Algorithm V and VI. Hence, Theorem 2.6 follows from Theorem 2.3-Theorem 2.5.

2.9.2. Non-quadratic Functions

The algorithms presented in Section 2.7 are defined also for nonquadratic functions. But the proofs of Theorems in the preceeding paragraph do not hold in general since quadratic properties of the function are used to prove the Theorems. At first we shall note that relations (i) in Theorem 2.1 do not hold for nonquadratic function in general, so that conjugate relations of vectors (s_0, s_1,..) do not hold for Algorithm (A) and (B). But relations $Y_i^T s_i=0$ (i=1, 2,...) are valid when Algorithm (C) is applied to nonquadratic functions as remarked in Section 2.7. Using this results it is also proved for Algorithm (C) that

$$y_i^T H_i R^{-1} H_i = y_i^T H_i ,$$

and that

$$s_i^T R^{-1} H_i = s_i .$$

Therefore the results in the preceeding paragraph are valid also when Algorithm (C) is applied to nonquadratic functions. That is,

Theorem 2.7

If $c_i = R^{-1}(\alpha_{1i} H_i^T y_i + \beta_{1i} s_i)$ and

$d_i = R^{-1}(\alpha_{2i} H_i^T y_i + \beta_{2i} s_i)$ in Algorithm (C),

and if vectors $(s_0, s_1, \ldots, s_i)$ are defined and non-zero, then

$$p_{i+1} = -H_{i+1}^T g_{i+1}$$

$$= \gamma_{i+1} q_{i+1},$$

$$q_{i+1} \equiv \{I_n - \frac{s_i y_i^T}{s_i^T y_i}\} H_i^T g_{i+1}$$

$$= \prod_{j=0}^{i} (I_n - \frac{s_i y_i^T}{s_i^T y_i}) R^T g_{i+1}.$$

Corollary

Assume that Algorithm X-XII are applied to nonquadratic functions and that sequences of minimizing points $(x_0, x_1, \ldots, x_i)$ are generated then they are identical if the initial point and the initial matrix R is the same for all algorithms.

Instead of Algorithm (A) and (B) we shall consider the

following two algorithms;

Algorithm (A')

$$H_{i+1} = H_i + \frac{s_i d_{1i}^T}{d_{1i}^T y_i} - \frac{(H_i - RK_i) y_i c_{1i}^T}{c_{1i}^T y_i}$$

$$-R\{\frac{y_i d_{2i}^T}{d_{2i}^T y_i} - \frac{(H_i - RK_i) y_i c_{1i}^T}{c_{1i}^T y_i}\}, \quad H_0 = R, \tag{2.78}$$

$$K_{i+1} = K_i - \{\frac{(K_i - I_n) y_i c_{2i}^T}{c_{2i}^T y_i} + \frac{y_i d_{2i}^T}{d_{2i}^T y_i}\}, \quad K_0 = I_n, \tag{2.79}$$

where

$$\begin{aligned} c_{ji} &= \alpha_{ji} H_i^T y_i + \beta_{ji} s_i, \quad (j=1,2) \\ d_{ji} &= \alpha_{ji} H_i^T y_i + \beta_{ji} s_i, \quad (j=1,2). \end{aligned} \tag{2.80}$$

Algorithm (B')

$$H_{i+1} = H_i + \frac{(s_i - H_i y_i) c_i^T}{c_i^T y_i} + \frac{c_i (s_i - H_i y_i)^T}{c_i^T y_i}$$

$$- \frac{c_i (s_i^T y_i - y_i^T H_i y_i) c_i^T}{(c_i^T y_i)^2}, \quad H_0 = R, \tag{2.81}$$

where

$$c_i = \alpha_i H_i^T y_i + \beta_i s_i. \tag{2.82}$$

These algorithms are obtained from Algorithm (A) and (B) respectively under the assumption of quadraticity of objective functions. For the methods defined above the same proposition as in Theorem 2.3 holds.

Theorem 2.8

There exist scalars γ_{i+1} and γ_{i+1}' such that

$$p_{i+1} = -H_{i+1}{}^T g_{i+1}$$
$$= \gamma_{i+1} q_{i+1}$$

for Algorithm (A'), and

$$p_{i+1} = -H_{i+1}{}^T g_{i+1}$$
$$= \gamma_{i+1}' q_{i+1}$$

for Algorithm (B') if H_i is symmetric, where

$$q_{i+1} \equiv \left(I_n - \frac{s_i y_i{}^T}{s_i{}^T y_i}\right) H_i{}^T g_{i+1} .$$

proof

In the proof of Theorem 2.3 quadraticity of objective functions are used only to show that

$$c_{1i}{}^T J_i = \alpha_{1i} y_i{}^T H_i + \beta_{1i} s_i{}^T,$$
$$c_{2i}{}^T K_i = \alpha_{2i} y_i{}^T H_i + \beta_{2i} s_i{}^T,$$
$$d_{2i}{}^T K_i = \alpha_{3i} y_i{}^T H_i + \beta_{3i} s_i{}^T,$$

$$(H_i - RK_i)^T g_{i+1} = 0,$$

for Algorithm (A) and that

$$c_i^T J_i = \alpha_i y_i^T H_i + \beta_i s_i^T,$$

$$H_i^T = H_i$$

for Algorithm (B).

Then the same calculation as in the proof of Theorem 2.3 are valid in this case, and we have for Algorithm (A')

$$\begin{aligned} p_{i+1} &= -H_{i+1}^T g_{i+1} \\ &= -\{1 - \frac{\alpha_{2i} y_i^T H_i^T g_{i+1}}{c_{2i}^T y_i} + \frac{(s_i^T y_i)(y_i^T R g_{i+1})(\alpha_{2i}\beta_{3i} - \alpha_{3i}\beta_{2i})}{(c_{2i}^T y_i)(d_{2i}^T y_i)} \\ &\quad + \frac{y_i^T (H_i - RK_i)^T g_{i+1} s_i^T y_i (\alpha_{2i}\beta_{1i} - \beta_{2i}\alpha_{1i})}{(c_{2i}^T y_i)(c_{1i}^T y_i)}\} \\ &\quad \times (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}, \end{aligned}$$

and for Algorithm (B'),

$$\begin{aligned} p_{i+1} &= -H_{i+1}^T g_{i+1} \\ &= -\{1 - \frac{(y_i^T H_i g_{i+1})}{(c_i^T y_i)} (\alpha_i^2 (y_i^T H_i y_i + s_i^T y_i)) + 2\alpha_i \beta_i s_i^T y_i\} \\ &\quad \times (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}. \end{aligned}$$

Corollary

If a symmetric matrix H_i is given, then (i+1)th search direction p_{i+1} generated by Algorithm I-IX are scalar multiplications of the vectors

$$q_{i+1} \equiv (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}.$$

proof

It is easily proved that Algorithm I -IX except VII are derived from Algorithm (A') and (B') by particular choices of parameters and the assumption that $RK_i y_i = H_i y_i$. Then the above Corollary 3 follows from Theorem 2.8 and Theorem 2.5, which is valid also for nonquadratic case.

One of well known rapidly convergent algorithms is Fletcher-Reeve's conjugate gradient method (Ref. 25). At the end of this section a comment concerning this algorithm will be given, according to Huang (Ref. 39) From Theorem 2.4, searching directions q_i of algorithms presented are expressed by formula (2.77), which can be rewritten in the form

$$q_0 = -R^T g_0$$

$$q_{i+1} = -\{R^T - \sum_{j=0}^{i} \frac{s_j y_j^T R^T}{y_j^T s_j}\}^T g_{i+1}$$

This algorithm is characteraized by the following updating

formula for matrices H_i (i=1, 2,...):

$$H_{i+1} = H_0 - \frac{Ry_i s_i^T}{s_i^T y_i} .$$

If initial matrix R=I and the function to be minimized is quadratic, then the above formula is symplified to

$$q_0 = -g_0$$

$$q_i = -g_i + \frac{g_i^T g_i}{g_{i-1}^T g_{i-1}} q_{i-1} .$$

This is Fletcher-Reeves's conjugate gradient algorithm. From the derivation of this algorithm, it is clear that all algorithms presented in Section 2.8, including Fletcher and Reeve's Conjugate Gradient Method, produce the same searching directions for quadratic function if initial matrix R=I.

2.10. Exactness of Algorithms

In a variable-metric algorithm, if the minimum of a positive definite quadratic form is attained at most n step, then we shall call the algorithm to be exact (Ref. 33, 38). From this definition and Theorem 2.2, any particular algorithms presented in Section 2.7 are exact if the vectors s_i do not vanish at the steps where $g_i \neq 0$. By definitions $s_i = \alpha_i p_i$, $\alpha_i = -\frac{(p_i, g_i)}{(p_i, Ap_i)}$ and $p_i = \gamma_i q_i$. From the Corollary of

Theorem 2.4, $q_i^T g_i = g_i^T R g_i$, so that $q_i \neq 0$ if R is positive definite and $g_i \neq 0$. Moreover $p_i^T g_i = \gamma_i q_i^T g_i \neq 0$ if $\gamma_i \neq 0$ and R is positive definite. Theorefore, given an algorithm, if the recursive formula for matrices H_i of the algorithm is well defined and $\gamma_i \neq 0$ at the step where $g_i \neq 0$, and if initial matrix R is of definite sign, then the algorithm is exact. Parameter γ_i's which determine the length of searching vectors p_i can be calculated from general formulas in the Appendix of this chapter. And they are summarized in the following table:

No. of Algorithms	$-\gamma_{i+1}$
I, IV, X	$1 - \dfrac{y_i^T H_i^T g_{i+1}}{y_i^T H_i y_i}$
II, IX, XI	1
III, XII	$1 + \dfrac{y_i^T H_i^T g_{i+1}}{(s_i - H_i^T y_i)^T y_i}$
V	$1 + \dfrac{y_i^T R g_{i+1}}{y_i^T H_i y_i}$
VI	$1 - \dfrac{y_i^T (H_i^T - R) g_{i+1}}{y_i^T H_i y_i}$
VII	$1 - \dfrac{y_i^T (H_i^T - 2R) g_{i+1}}{2(y_i^T H_i y_i)}$
VIII	$1 - \dfrac{(y_i^T H_i^T g_{i+1})(y_i^T H_i y_i + s_i^T y_i)}{(y_i^T H_i y_i)}$

For Algorithm II, IX, XI, γ_{i+1} is constant and $y_i^T H_i y_i = y_i^T R y_i$, so that these Algorithms are exact if R is positive definite. For Algorithm I IV X, assume $\gamma_i \neq 0$, then

$$-\gamma_{i+1} = \frac{g_i^T H_i^T g_i}{y_i^T H_i y_i} = \frac{\gamma_i g_i^T q_i}{y_i^T H_i y_i} .$$

Hence $\gamma_{i+1} \neq 0$ if $y_i^T H_i y_i \neq 0$ and R is positive definite. It is known that matrices H_i (i=1,2,...) in AlgorithmIV are always positive definite if R is positive definite, so that $y_i^T H_i y_i \neq 0$ and Algorithm IV is exact. For other algorithms, detailed considerations are not made. In concluding this section some discussions concering stability of the algorithms are given. If a symmetric i-th variable-metric matric matrix is given, the algorithms I-XII are equivalent with respect to search directions at (i+1)-th step from Theorem 2.7 and 2.8 . But from the computational point of view some differences are exist. At first singularity of matrices H_i (i=1,2,.) is to be avoided. And symmetricity of matrix H_i is desirable. Hence, we shall discuss Algorithm III,IV,VIII, and IX which belong to Algorithm (A) and (B) and generate symmetric variable-metric matrices if initial matrix R is symmetric. Let us write these recursive formulas of H_i as the following forms for symplicity;

$$H_{i+1}^{j} = H_i + E_i^{j} \qquad (j=3,4,8,9;\ i=1,2,\ldots).$$

For these symmetric variable-metric matrices the following results are obtained.

Theorem 2.9

Suppose that searching points x_i and x_{i+1} and an (i-th) variable metric matrix H_i are given. If

$$\gamma_i = \frac{y_i^T s_i}{y_i^T H_i y_i} > 0,$$

then

(1) for $\gamma_i > 1$,

$$H_{i+1}^8 \leq H_{i+1}^4 \leq H_{i+1}^3 \leq H_{i+1}^9,$$

(2) for $\gamma_i < 1$

$$H_{i+1}^3 \leq H_{i+1}^8 \leq H_{i+1}^4 \leq H_{i+1}^9,$$

(3) for $\gamma_i = 1$

$$H_{i+1}^8 \leq H_{i+1}^4 \leq H_{i+1}^9.$$

proof

It is known that $E_i^9 \geq E_i^4$, and that $E_i^8 = \gamma_i E_i^4 + (1-\gamma_i) E_i^3$, and $E_i^9 = \frac{1}{\gamma_i} E_i^4 + (1 - \frac{1}{\gamma_i}) E_i^3$ for $\gamma_i \neq 1$ (Ref. 38). Therefore when $\gamma_i \neq 1$ we have the relations;

$$E_i^9 - E_i^4 = (\frac{1}{\gamma_i} - 1)(E_i^4 - E_i^3),$$

$$E_i^8 - E_i^9 = (\gamma_i - \frac{1}{\gamma_i})(E_i^4 - E_i^3),$$

$$E_i^8 - E_i^4 = (\gamma_i - 1)(E_i^4 - E_i^3),$$

$$E_i^8 - E_i^3 = \gamma_i (E_i^4 - E_i^3).$$

And when $\gamma_i=1$ we have;

$$E_i^4 = 1/2(E_i^9+E_i^8).$$

From these relations and the property that $E_i^9 \geq E_i^4$ the propositions in Theorem 2.9 are easily obtained.

Q.E.D.

It is known that Algorithm IV(DFP Method) is stable (i.e. $H_i>0$) if initial matrix is positive definite. Then from Theorem 2.9, Algorithm XI is also stable and may be expected to be superior to DFP Method in the stability point of view.

2.11. Numerical Examples

In this section some results of numerical experiments are presented.

2.11.1 Test Problems

(i) Quadratic Function:

$$f(x_1, x_2) = x_1^2-2x_1x_2+2x_2^2 \equiv 1/2x^TAx$$

For this function

$$A = \begin{pmatrix} 2 & -2 \\ -2 & 4 \end{pmatrix} \quad \text{and} \quad A^{-1} = \begin{pmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}.$$

(ii) Rosenbrock's Function:

$$f(x_1, x_2) = 100(x_2-x_1^2)^2+(1-x_1)^2$$

This function is one of the most popular test functions and has a minimum of 0 at $(x_1, x_2)=(1, 1)$. This function has a steep valley along the curve $x_2=x_1^2$.

(iii) Enzyme Function (Ref. 47)

$$f(x_1,x_2,x_3,x_4) = \sum_{i=1}^{11} \left| V_i - \frac{x_1(y_i^3+x_2y_i)}{y_i^2+x_3y_i+x_4} \right|^2 .$$

The parameters y_i, V_i are given below in table 2.1. This function attains its minimum $f=3.075\times10^{-4}$ at the point

$$x=(0.1928,\ 0.1916,\ 0.1234,\ 0.1362)$$

Table 2.1

i	V_i	y_i
1	.1954	4
2	.1947	2
3	.1735	1
4	.1600	.5
5	.0844	.25
6	.0627	.167
7	.0456	.125
8	.0342	.1
9	.0323	.0833
10	.0235	.0714
11	.0246	.0625

(iV) A Constrained Problem (Beale's Problem):

Minimize the function,

$$f(x_1,x_2,x_3)=9-8x_1-6x_2-4x_3+2x_1^2+2x_2^2+x_3^2+2x_1x_2+2x_1x_3$$

subject to

$$x_1\geq 0,\quad x_2\geq 0,\quad x_3\geq 0 \quad \text{and} \quad x_1+x_2+2x_3\leq 3.$$

The solution is $f=\frac{1}{9}$ at $x=(\frac{4}{3},\ \frac{7}{9},\ \frac{4}{9})$. This problem is transformed by SUMT (Ref.15) to the minimization of a function $T(x,\ \gamma)$ with parameter γ:

$$T(x,\gamma) \equiv f(x_1,x_2,x_3) + \gamma\left(\frac{1}{x_1} + \frac{1}{x_2} + \frac{1}{x_3} + \frac{1}{3-x_1-x_2-2x_3}\right).$$

And it is expected that minimum points of $T(x,\ \gamma)$ tend to that of the original problem if the parameter γ tends to zero.

2.11.2 Minimization of a Function on a Line

In every steps of minimization algorithms step-sizes α_i $(i=0,\ 1,\ldots)$ are to be determined so that

$$f(x_i+\alpha_i p_i) = \min_{\alpha} f(x_i+\alpha p_i).$$

This problem is called "minimization problem of a function on a line" or "Linear search problem". Several techniques such as interpolation method, method using Fibbonachi Numbers, or Golden-Cut method,are known for this problem. In the computations of the examples, Golden-Cut method with some

modifications is used (Ref. 48). Linear search problem is to determine α_i such that $y'(\alpha_i)=0$, where $y(\alpha)=f(x_i+\alpha p_i)$ and y' denotes $dy/d\alpha$. At first an interval which contains α_i is to be estimated. This is done following R. Fletcher (Ref. 25). That is, a tentative step length h is determined as follows;

$$h=k, \quad \text{if } 0<k<(p_i^T p_i)^{-\frac{1}{2}}$$

$$=(p_i^T p_i)^{-\frac{1}{2}}, \quad \text{otherwise,}$$

where

$$k=2(est-f_i)p_i^T g_i$$

and est is an estimation of the minimum of f(x). Then, $y'=p_i^T g(x_i+\alpha p_i)$ is examined at the points $\alpha=0$, h, 2h, 4h,..., a, b, with α doubled at each time, and b is the first of these value where either y' becomes nonnegative or y does not decrease. Then α_i is supposed to be in the interval $a\leq\alpha\leq b$.

At the next step Golden-Cut method is used to determine α_i.

The method is summarized as follows

(1) At each step interval (x*, x**) is considered.

(2) At k-th step, $x_k=x^*+\frac{1}{\tau^2}(x^{**}-x^*)$ and the value $f(x_k)$ is computed.

(3) If $f(x^*)\leq f(x_k)$, $x_k\rightarrow x^{**}$, $k+1\rightarrow k$ and return to (2).

(4) If $f(x^*)>f(x_k)$, $k+1\rightarrow k$ and go to the next step.

(5) $x_1=x^*+\frac{1}{\tau^2}(x^{**}-x^*)$

$x_2=x^*+\frac{1}{\tau}(x^{**}-x^*)$,

And if $f(x_1) \geq f(x_2)$, then $x_1 \to x^*$,

if $f(x_1) < f(x_2)$ $x_2 \to x^{**}$

and $k+1 \to k$ return to the step (2). In the steps 2 and 5 τ is a positive root of the equation

$$t^2 - t - 1 = 0.$$

The iteration is stopped if

$$|x^* - x^{**}| < \varepsilon_1 ,$$

where ε_1 is a given small value.

2.11.3 Resetting and Stopping Conditions

In every algorithms if denominators in the recursive formulas of matrices H_i become too small the algorithms are resetted, that is if $|y_i^T H_i y_i| \leq \varepsilon_2$ or $(s_i - H_i y_i)^T y \leq \varepsilon_2$. Then H_{i+1} is resetted as H_0 and the algorithms are repeated again. Iterations are stopped when

$$g_i^T g_i \leq \varepsilon_3,$$

where ε_3 is a prescribed small value. In the following computations initial matrix H_0 is always set as a unit matrix.

2.11.4 Computed Results

(i) Quadratic Function:

This problem is chosen to show the exactness of the algorithms. The Algorithms I-X are applied to the function. Almost the same results are obtained by the methods I-IX. For example, results after two iterations are shown in Table

2.2. In the second column of the table only the figures which are independent of the choices of the methods are written. In the fourth column the elements of the variable-metric matrices $H=(h_{ij})$ after two iterations are written. From this table we can see that all algorithms except Algorithm X give almost the same results and that the corresponding variable-metric matrices H_i tend to A^{-1}. The results by Algorithm X is different from those by the other methods and H_i tends to zero matrix. Theoretically, the identical sequences of vectors and values of the function are expected from all of these methods, so the differences in the table are owing to inexactness in the linear-searchings or round-off errors in computations. Experiments by Algorithms XI and XII are not made, but it seems that they show behaviors similar to that by Algorithm X.

Table 2.2 Quadratic Function
$x_0=(3.0,9.0), f_0=117, \varepsilon_f=10^{-4}, est=0.1$

Method	x	f(x)	H_2
I – IX	$x_1=4\times10^{-5}$ $x_2=-1\times10^{-5}$	3×10^{-9}	$h_{11}=1.0000$ $h_{21}=0.5000$ $h_{12}=0.5000$ $h_{22}=0.5000$
X	$x_1=-3.3316\times10^{5}$ $x_2=1.1482\times10^{-4}$	3.5128×10^{-8}	$h_{11}=2.47\times10^{-8}$ $h_{21}=1.86\times10^{-8}$ $h_{12}=1.86\times10^{-8}$ $h_{22}=-5.58\times10^{-9}$

(ii) Rosenbrock's Function:

In Table 2.3 and 2.4 the computed results are shown for the three different initial points. The applied methods are denoted in the first column, where CG-1 and CG-2 mean respectively Fletcher-Reeves's conjugate gradient method with resettings after each three iterations and the conjugate gradient method without resetting, and ST means the steepest descent method. From the third to fifth column the numbers of itterations required to reach the values of functions in the second column are shown. From these table we can see that by all of the algorithms presented in the preceeding sections the extremum point is obtained and the rates of the convergence are almost the same for Algorithms I-IX. Algorithm X gives a little slow convergence compared with the other methods, but the rate of convergence is comparable with Fletcher-Reeves's conjugate gradient method with resettings. The much improvement concerning the rate of convergence is made by resetting in the conjugate-gradient method. It is remarked that $p_i^T g_i > 0$ at some steps in Algorithms I, III, V, VI, VIII and X. In such cases the linear searchings are made in the direction of $-p_i$ in these computations. Number of steps required in linear searchings are about 25 for Algorithms I IX and 30 for Algorithm X. It is interesting and important to see the effect of the accuracy in linear searchings to the rates of convergence of the methods An example of the computations with another value of ε_1 is presented in Table 2.5. The comparison of this table with Table 2.4 and 2.5

shows that the accuracy in linear searchings has not much effect on the rate of convergence for Algorithms I-X. It is also noted that in the computations for Table 2.6 unstable steps, that is steps, where $p_i^T g_i > 0$, appeared in Algorithms I, III, V, VI, VIII and X.

It is shown theoretically and experimentally that matrices H_i, $(i=0,1,\ldots)$ tend to the inverse matrix of the Hessian matrix when the objective function is a positive definite quadratic form. But for nonquadratic functions the behaviour of H_i for i large are not known theoretically. In Table 2.6 the values of elements of matrices H_i at the converged steps are shown.

For Rosenbrock's function,

$$A \equiv (\partial f/\partial x_i \partial x_j)_{(1,1)} = \begin{pmatrix} 802, & -400 \\ -400, & 200 \end{pmatrix}$$

and $$A^{-1} = \begin{pmatrix} 0.5, & 1.0 \\ 1.0, & 2.005 \end{pmatrix}.$$

From Table 2.7 we can see that the variable-metric matrices H_i in Algorithms I-IX give good approximations of the inverse of Hessian matrix at the extremum point; x=(1,1). This result is very important and to be investigated theoretically.

Table 2.3 Rosenbrock's Function

est=0.1, $\varepsilon_1=10^{-6}$, $\varepsilon_2=10^{-10}$, $\varepsilon_3=10^{-8}$

Method	f(x)	Initial Point		
		$x_{10}=3.0$ $x_{20}=9.0$ $f_0=4.0$	$x_{10}=-1.2$ $x_{20}=1.0$ $f_0=24.2$	$x_{10}=0.0$ $x_{20}=1.0$ $f_0=101$
I	1.0	6	8	1
	1.0×10^{-2}	14	16	9
	1.0×10^{-6}	18	19	13
	1.0×10^{-10}	19	20	14
II	1.0	5	8	1
	1.0×10^{-2}	16	14	9
	1.0×10^{-6}	18	16	13
	1.0×10^{-10}	19	18	14
III	1.0	8	8	1
	1.0×10^{-2}	17	17	9
	1.0×10^{-6}	19	20	14
	1.0×10^{-10}	21	22	15
IV	1.0	8	8	1
	1.0×10^{-2}	17	16	9
	1.0×10^{-6}	20	18	13
	1.0×10^{10}	21	19	14
V	1.0	9	8	1
	1.0×10^{-2}	15	17	9
	1.0×10^{-6}	17	19	13
	1.0×10^{-10}	17	21	14
VI	1.0	8	8	1
	1.0×10^{-2}	16	17	9
	1.0×10^{-6}	20	19	13
	1.0×10^{-10}	21	21	14
VII	1.0	6	8	1
	1.0×10^{2}	14	14	9
	1.0×10^{-6}	17	17	13
	1.0×10^{-10}	19	18	14

Table 2.4 Rosenbrock's Function

est=0.1, $\varepsilon_1=10^{-6}$, $\varepsilon_2=10^{-10}$, $\varepsilon_3=10^{-8}$

Method	f(x)	Initial Point		
		$x_{10}=3.0$ $x_{20}=9.0$ $f_0=4.0$	$x_{10}=-1.2$ $x_{20}=1.0$ $f_0=24.2$	$x_{10}=0.0$ $x_{20}=1.0$ $f_0=101$
VIII	1.0	7	7	1
	1.0×10^{-2}	16	17	10
	1.0×10^{-6}	20	22	13
	1.0×10^{-10}	21	23	15
IX	1.0	8	8	1
	1.0×10^{-2}	16	14	9
	1.0×10^{-6}	20	16	13
	1.0×10^{-10}	21	17	14
X	1.0	7	7	1
	1.0×10^{-2}	21	19	14
	1.0×10^{-6}	27	24	20
	1.0×10^{-10}	28	28	22
CG-1	1.0	13	11	1
	1.0×10^{-2}	25	21	11
	1.0×10^{-6}	30	24	15
	1.0×10^{-10}	>32	28	>20
CG-2	1.0	30	44	1
	1.0×10^{-2}	72	59	11
	1.0×10^{-6}	76	103	16
	1.0×10^{-10}	>82	106	>22
ST	1.0	14	37	
	1.0×10^{-2}	54	>80	
	1.0×10^{-6}	64		
	1.0×10^{-10}			

Table 2.5 Rosenbrock's Function

$x_0=(3.0, 9.0)$, $f_0=4.0$

est=0.1, $\varepsilon_1=10^{-4}$, $\varepsilon_2=10^{-10}$, $\varepsilon_3=10^{-8}$

Method	f(x)			
	1.0	1.0×10^{2}	1.0×10^{-6}	1.0×10^{-10}
I	6	14	17	18
II	6	15	19	20
III	5	14	17	18
IV	6	14	18	19
V	9	17	20	21
VI	8	19	22	24
VII	6	15	19	20
VIII	7	17	21	22
IX	6	15	19	20
X	7	15	18	19
CG-1	28	45	48	>50
CG-2	63	89	97	102

Table 2.6 Approximations of A^{-1}

$x_0=(3.0, 9.0)$
est=0.1, $\varepsilon_1=10^{-6}$, $\varepsilon_2=10^{10}$, $\varepsilon_3=10^{-8}$

Method	H_i			
	h_{11}	h_{12}	h_{21}	h_{22}
I	0.499	0.999	1.000	2.006
II	0.512	1.022	1.022	2.047
III	0.435	0.871	0.871	1.749
IV	0.498	0.996	0.996	1.995
V	0.498	0.997	0.994	1.996
VI	0.499	0.998	0.990	1.984
VII	0.496	0.991	0.992	1.987
VIII	0.401	0.807	0.807	1.628
IX	0.500	0.999	0.999	2.003

(iii) Enzyme Function

Computed results are presented in Table 2.7. In the table the numbers in the colum of Iter. are those of required iterations until the stopping condition is satisfied, and the values of x and f(x) in the table are those at the converged points. All of presented algorithms succeeded to reach the extremumum point and the required numbers of iterations are less than those by Fletcher and Reeves's conjugate gradient method with resetting. In this example Algorithm I required more iterations compared with the other methods.This fact is seen in the computations with other initial points In this example unstable steps, that is, steps where $p_i^T g_i > 0$, appeared in Algorithms I, III, V, VI, VII, VIII and X. For this test function it is also verified experimentally that the rate of the convergence of presented algorithms is not sensitive to the accuracy in linear searchings.

In the computations of the examples (i)-(iii) resettings did not occur in all of the algorithms except in Algorithm X.

In the table CG-1 denote the conjugate gradient method with resetting after each 5 iterations.

Table 2.7 Enzyme Function

$x_0=(0.5,\ 1.0,\ -1.0,\ 0.5)$, $f_0=6.7635$

est=0.1, $\varepsilon_1=10^{-4}$, $\varepsilon_2=10^{-10}$, $\varepsilon_3=10^{-8}$

Method	Iter.	x				f(x) (10^{-4})
		x_1	x_2	x_3	x_4	
I	37	0.19264	0.19572	0.12449	0.13799	3.0754
II	19	0.19275	0.19187	0.12297	0.13631	3.0751
III	19	0.19279	0.19152	0.12313	0.13616	3.0751
IV	25	0.19281	0.19126	0.12303	0.13605	3.0751
V	21	0.19280	0.19130	0.12305	0.13607	3.0751
VI	20	0.19279	0.19152	0.12306	0.13617	3.0751
VII	19	0.19280	0.19132	0.12305	0.13608	3.0751
VIII	19	0.19280	0.19136	0.12307	0.13609	3.0751
IX	19	0.19280	0.19132	0.12304	0.13607	3.0751
X	15	0.19280	0.19146	0.12319	0.13613	3.0751
CG-1	38	0.19190	0.21597	0.13046	0.14686	3.0853
CG-2	150	0.20217	0.24133×10^{-1}	0.10508	$0\ 54018\times10^{-1}$	4.3841

(iv) A Constrained Problem

Algorithms I-X and Fletcher and Reeves's conjugate gradient method are applied. All of the Algorithms succeeded to attain the extremum point;

$$x^* \equiv (\frac{4}{3}, \frac{7}{7}, \frac{4}{9}) = (1.333..., 0.777..., 0.444...).$$

Algorithm I-IX showed almost the same behaviours and the computations are stopped for these methods after about 30 iterations. In this example iterations are stopped when the inequality, $|T_i - T_{i+1}|<\delta$ is satisfied, where δ is a given small number. The results for an initial point are presented in Table 2.8. The prescribed parameters are the followings;

$$\varepsilon_1=10^{-6},\ \varepsilon_2=10^{-10},\ \delta=10^{-8},\ est=10^{-8}\ .$$

In the table results by Algorithms IX, X and the conjugate gradient method without resetting are shown. The numbers in the colmun of Iter. are the total numbers of the itterations required to reach the corresponding values of x or f(x).

In this example resettings of the search-directions appeared in the methods II and III.

Table 2.8 A Constrained Problem

$x_0=(0.1, 0.1, 0.1)$, $f_0=7.29$, $T_0=37.67$

Algorithm IX

γ	x_1	x_2	x_3	$T(x,\gamma)$	$f(x)$	Iter.
1	0.8952	0.7053	0.4286	7.4158	0.7039	8
10^{-2}	1.3796	0.7375	0.3515	0.2598	0.1549	16
10^{-4}	1.3400	0.7731	0.4330	0.1202	0.1158	22
10^{-6}	1.3341	0.7773	0.4432	0.1121	0.1116	29

Algorithm X

1	0.8952	0.7052	0.4286	7.4158	0.7039	13
10^{-2}	1.3795	0.7374	0.3517	0.2598	0.1549	27
10^{-4}	1.3402	0.7732	0.4328	0.1210	0.1158	37
10^{-6}	1.3342	0.7772	0.4432	0.1121	0.1158	46

C-G Method

1	0.8952	0.7053	0.4286	7.4158	0.7338	11
10^{-2}	1.3793	0.7374	0.3518	0.2598	0.1549	23
10^{-4}	1.3402	0.7731	0.4328	0.1210	0.1158	34
10^{-6}	1.3338	0.7773	0.4433	0.1121	0.1158	56

2.12 Conclusions

There are many algorithms which belong to so-called "variable Metric Method" These algorithms can be derived from three generalized algorithms. An important property of these method is that they minimize a positive-definite, quadratic function of n variables in n steps if n iterations are excuted. And the n-th variable-metric matrices are equal to the inverse of the Hessian matrix of the quadratic form in Algorithms (A) and (B) and to a zero matrix in Algorithm (C) It is shown that the minimizing sequence by these algorithms is unique for quadratic functions. That is, given an initial estimate x_0 of an extremum point and a matrix R which determines an initial search direction from x_0, a sequence of minimizing points $(x_0, x_1, \ldots)$ is unique, if it is defined in fact, independently of particular algorithms. But it is another problem whether an extremum point of the given quadratic function is reached by all of these algorithms at most after n-times iterations from any estimation. Some algorithms may stop at non-extremum point or may not be defined at certain steps. The same uniqueness property hold for particular algorithms derived from Algorithm (C) also in the case of non-quadratic functions. Although the uniqueness of the minimizing sequence for a given initial point and initial matrix in Algorithms (A) and (B) may be expected also for non-quadratic functions from numerical experiments in the section 2.11 or form other examples (Ref. 40), the property

cannot be proved. If i-th and (i+1)-th points and i-th variable-metric matrix H_i are given (i+1)-th searching direction is uniquely determined independently of particular choices of algorithms in this paper. But from computational point of view positive definiteness of matrix H_{i+1} is desirable to generate searching directions. In the section 2.10 positivity of matrices H_i in the various methods is discussed.

The particular algorithms presented are applied to four typical problems. From the numerical results the following observations are made:

(i) Algorithms I-X are all successful in solving the problems.

(ii) Algorithms I-X yield faster convergence than Fletcher and Reeves's conjugate gradient method with resetting. Algorithms I-IX are almost equivalent with respect to the rate of convergence and superior to Algorithm X in convergence.

(iii) The rate of convergence is not sensitive to the accuracy in the linear-searchings.

(iv) Directions p_i are not always descent-directions in Algorithms I, III, V, VI, VIII and X.

From the practical point of view it is desirable that matrix H_i (i=0, 1,...) are symmetric and that inequalities; $p_i^T g_i<0$ (i=0, 1,...), are always valid. From the above observations Algorithm IV (DFP Method) and Algorithm IX (GoldFarb's Method) seems to be most promising.

Appendix (Proof of Theorem 2.3)

(i) Algorithm (A)

At first, using the conjugate property of vectors s_j $(j \geq 0)$ we can prove that

$$c_{1i}^T J_i = \alpha_{1i} y_i^T H_i + \beta_{1i} s_i^T, \tag{A-1}$$

$$c_{2i}^T K_i = \alpha_{2i} y_i^T H_i + \beta_{2i} s_i^T, \tag{A-2}$$

and

$$d_{2i}^T K_i = \alpha_{3i} y_i^T H_i + \beta_{3i} s_i^T. \tag{A-3}$$

$$H_{i+1}^T g_{i+1} = H_i^T g_{i+1} + \frac{J_i^T d_{1i} s_i^T g_{i+1}}{d_{1i}^T J_i y_i} - \frac{J_i^T c_{1i} y_i^T (H_i - RK_i)^T g_{i+1}}{c_{1i}^T J_i y_i}$$

$$- \{\frac{K_i d_{2i} y_i^T}{d_{2i}^T K_i y_i} - \frac{K_i c_{2i} y_i^T (I_n - K_i^T)}{c_{2i}^T K_i y_i}\} R^T g_{i+1}$$

$$= \{H_i^T g_{i+1} - \frac{K_i^T c_{2i} y_i^T H_i^T g_{i+1}}{c_{2i}^T K_i y_i}\}$$

$$+ (y_i^T R^T g_{i+1}) \{\frac{K_i^T c_{2i}}{c_{2i}^T K_i y_i} - \frac{K_i^T d_{2i}}{d_{2i}^T K_i y_i}\}$$

$$+ y_i^T (H_i - RK_i)^T g_{i+1} \{\frac{K_i^T c_{2i}}{c_{2i}^T K_i y_i} - \frac{J_i^T c_{1i}}{c_{1i}^T J_i y_i}\}.$$

Now we shall calculate each terms in the above formula, substituting parameters in (A-1)–(A-3). Then we get the following results;

$$A \equiv \{H_i^T g_{i+1} - \frac{K_i^T c_{2i} y_i^T H_i^T g_{i+1}}{c_{2i}^T K_i y_i}\}$$

$$= (1 - \frac{\alpha_{2i} y_i^T H_i^T g_{i+1}}{(c_{2i}^T K_i y_i)})(I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1} \qquad (A\text{-}4)$$

$$B \equiv (y_i^T R g_{i+1})\{\frac{K_i^T c_{2i}}{c_{2i}^T K_i y_i} - \frac{K_i^T d_{2i}}{d_{2i}^T K_i y_i}\}$$

$$= \frac{(s_i^T y_i)(y_i^T R g_{i+1})(\alpha_{2i}\beta_{3i} - \alpha_{3i}\beta_{2i})}{(c_{2i}^T K_i y_i)(d_{2i}^T K_i y_i)}(I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1} \qquad (A\text{-}5)$$

$$C \equiv y_i^T (H_i - RK_i)^T g_{i+1}\{\frac{K_i^T c_{2i}}{c_{2i}^T K_i y_i} - \frac{J_i^T c_{1i}}{c_{1i}^T J_i y_i}\}$$

$$= \frac{y_i^T (H_i - RK_i)^T g_{i+1} s_i^T y_i (\alpha_{2i}\beta_{1i} - \beta_{2i}\alpha_{1i})}{(c_{2i}^T K_i y_i)(c_{1i}^T J_i y_i)} (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1} \qquad (A\text{-}6)$$

Since for quadratic functions $(H_i - RK_i)^T g_{i+1} = 0$,

$$H_{i+1}^T g_{i+1} = A + B$$

$$= \{(1 - \frac{2i y_i^T H_i^T g_{i+1}}{c_{2i}^T K_i y_i})$$

$$+ \frac{(s_i^T y_i)(y_i^T R g_{i+1})(\alpha_{2i}\beta_{3i} - \alpha_{3i}\beta_{2i})}{(c_{2i}^T K_i y_i)(d_{2i}^T K_i y_i)}\} q_{i+1} ,$$

where $q_{i+1} = (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}$.

(ii) Algorithm (B)

In this case $c_i^T J_i = \alpha_i y_i^T H_i + \beta_i s_i^T$.

Then

$$H^T_{i+1} g_{i+1} = H_i g_{i+1} + \frac{(s_i - H_i y_i) c_i^T J_i g_{i+1}}{c_i^T J_i y_i}$$

$$- \frac{J_i^T c_i y_i^T H_i g_{i+1}}{c_i^T J_i y_i} - \frac{J_i^T c_i (s_i^T y_i - y_i^T H_i y_i) c_i^T J_i g_{i+1}}{(c_i^T J_i y_i)^2}$$

$$= H_i g_{i+1} - \frac{J_i^T c_i y_i^T H_i g_{i+1}}{c_i^T J_i y_i}$$

$$+ \{\frac{(s_i - H_i y_i) c_i^T J_i g_{i+1}}{c_i^T J_i y_i} - \frac{J_i^T c_i (s_i^T y_i - y_i^T H_i y_i) c_i^T J_i}{(c_i^T J_i y_i)^2}\}$$

$$= H_i g_{i+1} + \{\frac{\alpha_i (y_i^T H_i g_{i+1})}{(c_i^T J_i y_i)} H_i g_i - \frac{\beta_i s_i y_i^T H_i g_{i+1}}{(c_i^T J_i y_i)}$$

$$- \frac{\alpha_i (y_i^T H_i g_{i+1})}{(c_i^T J_i y_i)} H_i g_{i+1}\} + \frac{\alpha_i (\alpha_i + \beta_i)}{(c_i^T J_i y_i)^2} \{(y_i^T H_i y_i) s_i y_i^T H_i g_{i+1}$$

$$- (y_i^T H_i g_{i+1}) s_i^T y_i H_i g_{i+1} + (y_i^T H_i g_{i+1}) s_i^T y_i H_i g_i\}$$

$$= \{H_i g_{i+1} - \frac{s_i y_i^T}{s_i^T y_i} H_i g_{i+1}\}$$

$$+ \{\frac{s_i y_i^T}{s_i^T y_i} H_i g_{i+1} - \frac{\beta_i s_i y_i^T}{c_i^T J_i y_i} H_i g_{i+1} + \frac{\alpha_i (\alpha_i + \beta_i)(y_i^T H_i y_i)}{(c_i^T J_i y_i)^2} s_i y_i^T H_i g_{i+1}$$

$$+ \{\frac{\alpha_i (y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)} H_i g_i + \frac{\alpha_i (\alpha_i+\beta_i)(y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)^2} s_i{}^T y_i H_i g_i\}$$

$$- \{\frac{\alpha_i (y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)} H_i g_{i+1} + \alpha_i (\alpha_i+\beta_i) \frac{(y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)^2} s_i{}^T y_i H_i g_{i+1}\}$$

$$\equiv \{H_i g_{i+1} - \frac{s_i y_i{}^T}{s_i{}^T y_i} H_i g_{i+1}\} + A + B + C$$

$$A \equiv \{\frac{1}{s_i{}^T y_i} - \frac{\beta_i}{c_i{}^T J_i y_i} + \frac{\alpha_i (\alpha_i+\beta_i)(y_i{}^T H_i y_i)}{(c_i{}^T J_i y_i)^2}\} s_i y_i{}^T H_i g_{i+1}$$

$$= \frac{(y_i{}^T H_i y_i)}{(c_i{}^T J_i y_i)^2 (s_i{}^T y_i)} \{\alpha_i{}^2 (y_i{}^T H_i y_i + s_i{}^T y_i) + 2\alpha_i \beta_i (s_i{}^T y_i)\} s_i y_i{}^T H_i g_{i+1}$$

$$B \equiv \{\frac{\alpha_i (y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)} + \frac{\alpha_i (\alpha_i+\beta_i)(y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)^2} s_i{}^T y_i\} H_i g_i$$

$$= \frac{1}{(c_i{}^T J_i y_i)^2} \{\alpha_i{}^2 (y_i{}^T H_i y_i + s_i{}^T y_i) + 2\alpha_i \beta_i s_i{}^T y_i\} (y_i{}^T H_i g_i) \frac{s_i y_i{}^T}{s_i{}^T y_i} H_i g_{i+1}$$

$$C \equiv - \{\frac{(y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)} + \frac{\alpha_i (\alpha_i+\beta_i)(y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)^2} s_i{}^T y_i\} H_i g_{i+1}$$

$$= \frac{(y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)^2} \{\alpha_i{}^2 (y_i{}^T H_i y_i + s_i{}^T y_i) + 2\alpha_i \beta_i s_i{}^T y_i\} H_i g_{i+1}$$

Therefore

$$H_{i+1} g_{i+1} = \{1 - \frac{(y_i{}^T H_i g_{i+1})}{(c_i{}^T J_i y_i)} (\alpha_i{}^2 (y_i{}^T H_i y_i + s_i{}^T y_i) + 2\alpha_i \beta_i s_i{}^T y_i\} q_{i+1}$$

$$q_{i+1} = (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i g_{i+1}$$

(iii) Algorithm (C)

$$H^T_{i+1} g_{i+1} = H_i^T g_{i+1} - \frac{H_i^T c_i y_i^T (H_i^T - R) g_{i+1}}{(c_i^T H_i y_i)} - \frac{H_i^T d_i y_i^T R_i^T g_{i+1}}{(d_i^T H_i y_i)} .$$

In this case by the definition of H_i and conjugate property of s_j $(j>0)$,

$$c_i^T H_i = \alpha_{1i} y_i^T H_i + \beta_{1i} s_i^T$$

$$d_i^T H_i = \alpha_{2i} y_i^T H_i + \beta_{2i} s_i^T .$$

By the same procedure as in the proof (i),

$$H^T_{i+1} g_{i+1} = (H_i^T g_{i+1} - \frac{H_i^T c_i y_i^T H_i^T g_{i+1}}{(c_i^T H_i y_i)})$$

$$+ (y_i^T R g_{i+1}) \{ \frac{H_i^T c_i}{(c_i^T H_i y_i)} - \frac{H_i^T d_i}{d_i^T H_i y_i} \}$$

$$= (1 - \frac{\alpha_{1i} (y_i^T H_i^T g_{i+1})}{(c_i^T H_i y_i)}) (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}$$

$$+ \frac{(s_i^T y_i)(y_i^T R g_{i+1})(\alpha_{1i}\beta_{2i} - \alpha_{2i}\beta_{1i})}{(c_i^T H_i y_i)(d_i^T H_i y_i)} (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}$$

$$= 1 - \frac{\alpha_{1i}(y_i^T H_i^T g_{i+1})}{(c_i^T H_i y_i)} + \frac{(s_i^T y_i)(y_i^T R g_{i+1})(\alpha_{1i}\beta_{2i} - \alpha_{2i}\beta_{1i})}{(c_i^T H_i y_i)(d_i^T H_i y_i)} q_{i+1}.$$

$$q_{i+1} = (I_n - \frac{s_i y_i^T}{s_i^T y_i}) H_i^T g_{i+1}$$

Q.E.D.

CHAPTER III

EXTENSIONS OF VARIABLE-METRIC METHOD TO A FUNCTION SPACE

3.1. Introduction

As presented in the preceeding chapter, several rapidly convergent methods of function minimization problems have been proposed in recent years. Among these the Fletcher and Reeves's conjugate gradient method (Ref. 25) and Davidon's method, which is also called DFP method sometimes (Ref. 26, 31), are most popular. The conjugate gradient method is very simple. DFP method is more complex, but it is known by experience that its convergence is superior to that of the conjugate gradient method. The steepest descent method and Newton's method have been extended to function spaces and applied to control problems by researches such as H. J. Kelley (Ref. 7), A. E. Bryson and W. F. Denham (Ref. 6), R. McGill and R. E. Kopp (Ref. 9). L. S. Lasdon, S. K. Mitter and A. D. Waren (Ref. 28), or J. F. Sinnot Jr. (Ref. 30) tried to apply the conjugate gradient method to optimal control problems.

In this chapter, extensions of Davidon's method and the Fletcher-Reeves's conjugate gradient method to Hilbert space are presented. The stability and convergence of the methods are shown in the case when the functionals to be minimized

are quadratic. The methods are applied to optimal control problems and numerical examples are given.

3.2 Formulation of the Problem

Let H be a (real) separable Hilbert space with inner product (f, g), $f, g \in H$. The norm of an element $f \in H$ is defined as $|f| = (f,f)^{\frac{1}{2}}$. Let A be a linear self-adjoint operator on H such that

$$m|f|^2 \leq (f,Af) \leq M|f|^2, \tag{3.1}$$

where

$$M = \underset{|f|\neq 0}{\text{Sup}} \frac{(f,Af)}{|f|^2}, \quad m = \underset{|f|\neq 0}{\text{Inf}} \frac{(f,Af)}{|f|^2}, \tag{3.2}$$

and $0<m\leq M$. Then the norm of A is equal to M:

$$|A| = M. \tag{3.3}$$

Since M is finite, A is a continuous operator From the condition (3.1), an inequality

$$|Af| \geq m|f| \tag{3.4}$$

holds. The inequality is a necessary and sufficient condition for the inverse operator A^{-1} of the self-adjoint operator A to be defined. The inverse A^{-1} satisfies the inequalities

$$|A^{-1}g| \leq \frac{1}{m}|g|,$$

$$|A^{1}g| \geq \frac{1}{M}|g|, \qquad g \in H. \tag{3.5}$$

Using Schwartz's inequality,

$$\frac{1}{M}|g|^2 \leq (g, A^{-1}g) \leq \frac{1}{m}|g|^2 . \tag{3.6}$$

Since A is self-adjoint, A^{-1} is also a self-adjoint operator. Let S(f) be a Frechet differentiable function on H. We call the operator F, which is defined by the formula;

$$\lim_{t \to 0} \frac{1}{t}\{S(f+th)-S(f)\} = (F(f),h) \tag{3.7}$$

for any $h \in H$, the gradient of the functional S:

$$F = \text{grad } S$$

Problem

Let $S(u) = \frac{1}{2}(u,Au) - (u,g)$ be a quadratic form on H, where A is a linear self-adjoint operator satisfying the condition (3.1). Find the u* which minimizes the functional S(u).

By the definition of the gradient of functionals, grad S(u) exists and is defined by

$$\text{grad } S(u) = Au - g. \tag{3.8}$$

The gradient of S(u) is denoted by g(u):

$$g(u) = Au - g. \tag{3.9}$$

Then the solution of the above problem is given by

$$u^* = u - A^{-1}g(u) \quad , \tag{3.10}$$

and

$$S(u^*) = -\frac{1}{2}(g, A^{-1}g).$$

In other words, u^* is obtained directly if we can make use of the gradients $g(u)$ and the inverse operator A^{-1}. But, in this problem, we assume that A^{-1} cannot be evaluted directly.

We shall give some comments about the more general case of quadratic functionals. Suppose that functionals are only non-negative, that is, $m \geq 0$ in the condition (3.1). In this case there exists an element u^* H such that

$$\sqrt{A}u^* = g \quad ,$$

if

$$\inf_u S(u) > -\infty.$$

And

$$\inf S(u) = -\tfrac{1}{2}|u^*|^2, \text{(Ref. 50)}.$$

In the above $\sqrt{A}$ is the root operator of A, which is uniquely determined for a non-negative self-adjoint operator

3.3 Algorithm of Davidon's Method

In this section we distinguich steps of iterations of algorithms by super-scripts of the letters Let i-th approximation of the solution of the problem be u^i; then the (i+1)st approximation is determined as follows using the

gradient at u^i:

Define $p^i \in H$ as

$$p^i = -K^i g^i,$$

where g^i is the gradient at u^i;

$$g^i = Au^i - g,$$

and K^i is an operator from H to H such that

$$(f, K^i f) > 0 \quad f \in H \text{ and } f \neq 0.$$

Then u^{i+1} is given by

$$u^{i+1} = u^i + \alpha^i p^i ,$$

where α^i is a constatn which minimizes a function of α;

$$S(u^i + \alpha p^i) = S(u^i) + \alpha(p^i, g^i) + \frac{\alpha^2}{2}(p^i, Ap^i).$$

By this definition,

$$\alpha^i = -\frac{(p^i, g^i)}{(p^i, Ap^i)} . \tag{3.11}$$

The operator K^i is modified at each step so that p^i becomes an eigen-element of $K^{i+1}A$. The algorithm of the computation is given as follows:

(i) Choose an initial estimation u^0 and identify K^0 with an positive operator such that

$$\alpha|f|^2 \leq (f, K^0 f) \leq \beta|f|^2, \quad f \in H \ (f \neq 0), \quad 0 < \alpha \leq \beta.$$

(ii) Evaluate the gradient g^i at u^i

(iii) Set $p^i = -K^i g^i$.

(iv) Set $u^{i+1} = u^i + \alpha^i p^i$,

where α^i is a constant such that

$$S(u^i+\alpha^i p^i) = \min_{\alpha} S(u^i+\alpha p^i).$$

(v) Set $y^i = (g^{i+1}-g^i)/\alpha^i$.

(vi) Set $q^i = K^i y^i$.

(vii) Set

$$p_N{}^i = \frac{p_i}{\sqrt{(p^i,y^i)}} \quad , \quad q_N{}^i = \frac{q^i}{\sqrt{(q^i,y^i)}} \ .$$

(viii) Define an operator K^{i+1} as follows:

$$K^{i+1}f = K^i f + (f,p_N{}^i)p_N{}^i - (f,q_N{}^i)q_N{}^i,$$

where f is an arbitrary element of H.

(iX) Set i=i+1 and repeat (ii) (viii).

By the definition of g^i,

$$g^{i+1} = Au^{i+1}-g$$

$$= A(u^i+\alpha^i p^i)-g$$

$$= g^i+\alpha^i Ap^i \ .$$

Hence, from the above definition (v),

$$y^i = Ap^i. \tag{3.12}$$

Substituting (3.12) into (p^i,y^i), and considering positivity of A, we show that $(p^i,y^i)>0$. In the following section we shall show that $(q^i,y^i)=(K^iy^i,\ y^i)>0$. Therefore the vectors ${p_N}^i$, ${q_N}^i$ are well-defined, so that the operator K^{i+1} is also defined.

Now, suppose that S(u) is not necessarily quadratic. By the definition of α^i,

$$(p^i,\ g^{i+1}) = 0. \tag{3.13}$$

Then

$$\begin{aligned}(p^i,\ y^i) &= \frac{1}{\alpha^i}\{(p^i,\ g^{i+1})-(p^i,\ g^i)\}\\ &= \frac{1}{\alpha^i}\ (K^ig^i,\ g^i).\end{aligned}$$

On assuming the positivity of K^i, we have

$$(p^i,\ g^i) = -(K^ig^i,\ g^i)<0 \qquad \text{for } g^i\neq 0,$$

so that $\alpha^i>0$ and $(p^i,\ y^i)>0$ if $g^i\neq 0$. Hence, if K^i is positive also for non-quadratic functionals, vectors ${p_N}^i$ and ${q_N}^i$ are defined as well. The positivity of K^i in the case of the non-quadratic form is noted in the Remark 1 in the following section.

3.4. Stability and Convergence of the Scheme

In this section, we shall show that the value of the functional to be minimized decreases at each step and searching points converge to the extremum point with this

method. The following two lemmas (Lemma 3.1-3.2) are direct extensions of the results in Ref. 26, and the proofs formally follow proofs in the reference.

Lemma 3.1

K^i is a linear self-adjoint, positive operator and $(f, K^i f)=0$ only if $f=0$ $(i=1, 2,...)$.

proof

We shall prove the lemma by induction. Since K^0 is positive, the assertion is trivial for $i=0$. Assume the lemma is valid for $i=1, 2,... n$; we shall now prove that the statement holds for $i=n+1$. From (viii) it is clear that K^{n+1} is a linear self-adjoint operator. Hence, it is sufficient to show positivity of K^{n+1}. From the relation (vi)-(viii),

$$(f,K^{n+1}f) = (f,K^n f)+(f,p_N{}^n)^2 - (f,q_N{}^n)^2$$

$$= \frac{(f,K^n f)(y^n,K^n y^n)-(f,K^n y^n)^2}{(y^n,K^n y^n)}$$

$$+(f,p_N{}^n)^2.$$

Since K^n is a positive operator, inequality

$$(f,K^n f)(y^n,K^n y^n)\geq(f,K^n y^n)^2$$

holds by Schwarz's inequality. Therefore the first term of the right-hand side of the above equality is nonnegative, and the second term is clearly nonnegative. The first term

becomes zero only if f is a scalar multiple of y^n:

$$f = \beta y^n$$

$$= \frac{\beta}{\alpha^n} (g^{n+1} - g^n) ,$$

where β is an arbitrary constat. From this fact and the relation (3.13), $(f, K^{n+1} f)$ vanishes if and only if $(g^n, p^n) = 0$. But this contradicts the positiveness of K^n. Hence, K^{n+1} is a positive operator and the lemma is proved.

Remark 1. In the above proof the quadratic property of the functional S(u) is not used. Therefore the assertion of Lemma 1 is valid also in nonquadratic functionals.

Lemma 3.2

The relations

$$(p_N{}^i, Ap_N{}^j) = \delta_{ij}, \quad i<k, \quad j<k, \tag{3.14}$$

$$K^k Ap_N{}^i = p_N{}^i, \quad i<k, \quad i=1,2,\ldots, \tag{3.15}$$

hold, where δ_{ij} is Kronecker's symbol.

proof

From (vi) and (viii)

$$K^{i+1} y^i = K^i y^i + (y^i, p_N{}^i) p_N{}^i \; (y^i, q_N{}^i) q_N{}^i$$

$$= K^i y^i + p^i - K^i y^i$$

$$= p^i .$$

Hence

$$K^{i+1}Ap^i = p^i \tag{3.16}$$

by (3.12). The statement of the Lemma is satisfied for k=1 by (3.16). Assume that the relation (3.14) and (3.15) are satisfied for k=n. From (3.8)

$$g^n = g^{i+1} + \sum_{j=i+1}^{n-1} \alpha^j Ap^j \quad 0 \leq i < n. \tag{3.17}$$

From relations (3.13)

$$(p^i, g^{i+1}) = 0, \quad i = 0, 1, 2, \ldots, n.$$

Hence, from (3.17), (3.12) and (3.14) with k=n.

$$(p^i, g^n) = (p^i, g^{i+1}) = 0. \tag{3.18}$$

Therefore

$$(K^n Ap^i, g^n) = (Ap^i, K^n g^n) = 0,$$

since (3.15) holds for k=n. Substituting $p^i = -K^i g^i$, we obtain a formula

$$(Ap^i, p^n) = 0, \quad 0 \leq i < n. \tag{3.19}$$

Now, by the self-adjointness of K^n and A ,

$$(K^n y^n, Ap^i) = (y^n, K^n Ap^i)$$

$$= (Ap^n, p^i), \quad 0 \leq i < n,$$

taking into consideration relations (3.12), (3.15) and (3.19). By using this result it is simple to prove the equalities

$$K^{n+1}Ap^i = K^nAp^i \qquad 0\leq i<n,$$
$$= p^i, \qquad (3.20)$$

by the definition of K^{n+1}. The relations (3.16), (3.19) and (3.20) show that the statement in the lemma hold for k=n+1.

Lemma 3.3

Let $\psi_i \in H$, (i=0,1,2,...) be a complete system in H, which satisfies conditions

(1) $(\psi_i, A\psi_j) = \delta_{ij}$,

(2) $m|f|^2 \leq (f,Af) \leq M|f|^2$, m, M>0.

Then, for any element $f \in H$, the following equalities hold:

$$f = \sum_{i=0}^{\infty} (f,\psi_i)A\psi_i$$
$$= \sum_{i=0}^{\infty} (f,A\psi_i)\psi_i .$$

proof

Denote $(f,A\psi_i)$ by d_i; then, we have the inequalities

$$M|f|^2 \geq (f,Af) \geq \sum_1^n d_i^2$$

since

$$(f-\sum_0^n d_i\psi_i,\ A(f-\sum_0^n d_i\psi_i)) = (f,Af)-\sum_0^n d_i^2 \geq 0.$$

Define f_n as

$$f_n = \sum_0^n d_i \psi_i ;$$

then

$$0 \leq m|f_s - f_t|^2 \leq (f_s - f_t,\ A(f_s - f_t)) = \sum_{t+1}^{s} d_i^2 \quad s \geq t.$$

The right-hand side of the equality tends to zero as t and s tend to infinity. Therefore, there exists an element $\psi \in H$ such that $f_n \to \psi$ as $n \to \infty$. The element is expressed as

$$\psi = \sum_0^\infty d_i \psi_i .$$

By the condition of the lemma,

$$(f-\psi,\ A\psi_i) = d_i - d_i = 0, \quad i=0,1,\ldots$$

Since $\{\psi_i\}$ is a complete system, the equalities mean that ψ is identical with f. Using A^{-1} in place of A in the above discussions, the last part of the lemma can be proved.

This lemma asserts that if $\{\psi_i\}$ is a complete system, then $\{A\psi_i\}$ is also a complete system. We introduce a well known property with respect to an increasing sequence of self-adjoint operators.

Lemma 3.4 (Ref. 49)

Let $\{u_i\}$ be an increasing sequence of positive self-adjoint operators such that

$$\sup_n |U_n| < A < +\infty .$$

Then, there is a linear operator U such that $Uf=\lim_{n\to\infty} U_n f$ for any $f\in H$, and $|U_n|\leq A$.

In the above lemma an increasing sequence of operators means a system of operators $\{U_n : n=0, 1,2,\ldots\}$ such that

$$(f, U_n f)\leq(f, U_{n+1} f)$$

for an arbitrary $f\in H$ and for $n=0, 1,\ldots$

Theorem 3.1

The sequence of operators $\{K^i\}$ is uniformly bounded and converges on H to a linear operator K.

proof

Denote by A_n, $n=0,1,2,\ldots$, an operator such that

$$A_n f = \sum_{i=0}^{n} (f, p_N{}^i) p_N{}^i \quad \text{for } f\in H.$$

The elements $p_N{}^i$, $i=0,1,2,\ldots, n$, satisfy the conditions of Lemma 3.3. Add a sequence r^i, $i=-1,12,\ldots$, to $p_N{}^i$ so that a system of elements $\{r^i, p_N{}^i : i=-1,\ldots, -n\ldots, j=0,1, 2,\ldots, n,\ldots\}$ becomes a complete system satisfying the conditions of Lemma 3.3. Then for any $f\in H$,

$$A^{-1} f = \sum_{i=0}^{n} (f, p_N{}^i) p_N{}^i + \sum_{i=-\infty}^{-1} (f, r^i) r^i,$$

by Lemma 3.3, so that

$$(f, A^{1}f) = \sum_{i=0}^{n} (f, p_N^{i})^2 + \sum_{i=-\infty}^{1} (f, r^{i})^2 \geq \sum_{i=0}^{n} (f, p_N^{i})^2.$$

The right-hand side of this inequality is equal to $(f, A_n f)$. Therefore

$$M'|f|^2 \geq (f, A^n f);$$

in other words, $|A_n| \leq M'$ where $M' = \frac{1}{m}$. $\{A_n\}$ is an increasing sequence of positive self-adjoint operators by the definition of A_n. Therefore, by Lemma 3.4, there exists a linear operator A such that

$$Af = \lim_{n\to\infty} A_n f, \quad f \in H,$$

and $|A| \leq M'$.

Now, define operators B_n, $n=0,1,\ldots$, as

$$B_n f = \sum_{i=0}^{n} (f, q_N^{i}) q_N^{i}.$$

Then the operator K^{n+1} is expressed as

$$K^{n+1} = I + A_n - B_n.$$

Hence, for an arbitrary $f \in H$,

$$(f, K^{n+1}f) = |f|^2 + (f, A_n f) - (f, B_n f).$$

Since K^{n+1} is a positive operator,

$$(f, B_n f) \leq |f|^2 + (f, A_n f) \leq (M'+1)|f|^2.$$

Hence B_n is bounded.

$$|B_n| \leq (M'+1), \quad n=0,1,2,....$$

Since B_n is also an increasing sequence, by Lemma s.4 there exists an operator B such that

$$Bf = \lim_{n\to\infty} B_n f \quad \text{for} \quad f \in H$$

and

$$|B| \leq (M'+1).$$

Let us define an operator K by

$$K=I+A-B.$$

Then, it is clear from the above discussions that K is a linear bounded operator such that

$$Kf = \lim_{n\to\infty} K^n f$$

and

$$|K| \leq 2(M'+1).$$

Hence, the theorem is proved.

We shall show that the values of the given functional decrease with each step.

Theorem 3.2

With the scheme defined in Section 3.3,

$$S(U^{i+1})<S(U^i) \quad \text{for} \quad g^i \neq 0, \quad i=0,1,2,....$$

proof

We shall show that the inner product of the direction of search p^i and the gradient g^i is negative and the step size α^i is positive for every i, i=0,1,2, .. . Since $p^i=-K^i g^i$ and K^i is positive from Lemma 3.1,

$$(p^i,g^i) = -(K^i g^i, g^i)<0 \quad \text{for} \quad g^i \neq 0, \; i=0,1,2,...$$

By the definition of α^i,

$$\alpha^i = \frac{(K^i g^i, g^i)}{(p^i, Ap^i)} ,$$

so that $\alpha^i>0$ for $g^i \neq 0$. From these considerations the statement of the theorem is valid.

Next, it will be shown that u^i converge to the extremum point u^* as $i \to +\infty$ and that there is a subspace of H on which the sequence of operators K^i converges to A^{-1}.

Lemma 3.5

$K^i y^i \in H$ is expressed as a linear combination of $K^0 Ap^j$, j=0,1,2,..., i.

proof

For i=0, the assertion of the lemma is valid since $K^0 y^0 = K^0 Ap^0$. It is assumed that the lemma holds for $K^j y^j$

(j=0,1,..., i). Then from (viii) in the algorithm defined in the section 3.3.

$$K^{i+1}y^{i+1} = K^0y^{i+1} - \sum_{j=0}^{i} \frac{(y^{i+1},K^jy^j)}{(y^j,K^jy^j)} K^jy^j.$$

Since $y^{i+1}=Ap^{i+1}$ from (3.12), the right-hand side of the above equality is a linear combination of K^0Ap^j, j=0,1,2,..., i+1.

Theorem 3.3

Let u^i, i=0,1,2,..., be a sequence of element in H as defined in the section 3.3; then, the sequence converges to u^* as $i\to\infty$.

proof

From (3.11)

$$S(u^{i+1}) = S(u^i) - \frac{(K^ig^i,g^i)^2}{(p^i,Ap^i)}$$

$$= S(u^i) - \frac{(g^i,K^ig^i)^2}{(K^ig^i,AK^ig^i)} .$$

Since $S(u^i)$ is bounded and monotone decreasing by Theorem 3.2,

$$\frac{(g^i,K^ig^i)^2}{(K^ig^i,AK^ig^i)} \to 0 \quad \text{as } i\to+\infty. \tag{3.21}$$

By Schwartz's inequality,

$$(K^i g^i,\ K^i g^i)^2 \le (K^i g^i,\ g^i)(K^i(K^i g^i), K^i g^i)$$

$$\le (K^i g^i,\ g^i)|K^i| \cdot |K^i g^i|^2$$

$$\le 2(M'+1)|K^i g^i|^2 (K^i g^i, g^i).$$

Hence,

$$(K^i g^i,\ g^i)^2 \ge \frac{|K^i g^i|^4}{4(M'+1)^2}.$$

From the condition (3.1) for A,

$$M|K^i g^i|^2 \ge (K^i g^i,\ AK^i g^i).$$

Combining the above two inequalities, we have

$$\frac{(g^i, K^i g^i)^2}{(K^i g^i, AK^i g^i)} \ge \frac{1}{4M(M'+1)^2} |K^i g^i|^2.$$

Since the left-hand side of this inequality tends to zero from (3.21),

$$|K^i g^i| \to 0 \qquad \text{as } i \to \infty. \tag{3.22}$$

From (3.6) and the condition for K^0,

$$m'|K^{0-1}K^i g^i|^2 \le (K^{0-1}K^i g^i, A^{-1}K^{0-1}K^i g^i) \le M'|K^{0-1}K^i g^i|^2$$

and

$$\frac{1}{\beta}|K^i g^i| \le |K^{0-1}K^i g^i| \le \frac{1}{\alpha}|K^i g^i|.$$

Hence

$$\frac{m'}{\beta^2}|K^i g^i|^2 \leq (K^{0-1}K^i g^i, A\ {}^1K^{0-1}K^i g^i) \leq \frac{M'}{\alpha^2}|K^i g^i|^2,$$

where

$$M' = \frac{1}{m} \text{ and } m' = \frac{1}{M}.$$

Therefore

$$(K^{0-1}K^i g^i, A\ {}^1K^{0-1}K^i g^i) \to 0 \quad \text{as } i \to +\infty. \tag{3.23}$$

From (3.11), (3.12) and (3.14),

$$\begin{aligned} g^i &= g^0 + \sum_{j=0}^{i-1} \alpha^j A p^j \\ &= g^0 - \sum_{j=0}^{i-1} (g^j, p_N{}^j) A p_N{}^j \\ &= g^0 - \sum_{j=0}^{i-1} (g^0 + \sum_{k=0}^{j-1} \alpha^k A p^k, p_N{}^j) A p_N{}^j \\ &= g^0 - \sum_{j=0}^{i-1} (g^0, p_N{}^j) A p_N{}^j, \end{aligned} \tag{3.24}$$

so that

$$(g^i, p_N{}^j) = 0 \qquad j<i.$$

On the other hand by (viii) of the Algorithm and (3.14),

$$K^i g^i = K^0 g^i - \sum_{j=0}^{i-1} \frac{(g^i, K^j g^j)}{(y^j, K^j y^j)} K^j y^j$$

The second term of the right-hand side of this equality is a linear combination of K^0Ap^k, $k=0,1,\ldots, i-1$. Hence,

$$K^{0-1}K^i g^i = g^i - \sum_{j=0}^{i\ 1} \beta_j{}^i Ap_N{}^j, \qquad (3.25)$$

where $\beta_j{}^i$, $j=0,1,\ldots, i\ 1$, are appropriate constants.
By Lemma 3.2,

$$(K^{0-1}K^i g^i,\ A^{-1}K^{0-1}K^i g^i)$$

$$= (g^i - \sum_{j=0}^{i-1} \beta_j{}^i Ap_N{}^j,\ A^{-1}(g^i - \sum_{j=0}^{i-1} \beta_j{}^i Ap_N{}^j))$$

$$= (g^i,\ A^{\ 1}g^i) + \sum_{j=0}^{i-1} (\beta_j{}^i)^2.$$

A^{-1} is a positive operator, and by (3.23)

$$(g^i A^{-1} g^i) \to 0 \quad \text{as} \quad i\to\infty.$$

Therefore, taking into consideration the inequalities

$$m'|g^i|^2 \leq (g^i, A^{-1}g^i),$$

we see that the gradient of $S(u^i)$ tends to zero as $i\to\infty$:

$$|g^i| \to 0 \quad \text{as} \quad i\to\infty.$$

By definition of g^i, this means that the sequence u^i, $i=0$, 1,2,. ., converges to u^*.

Theorem 3.4

There is a subspace $\overline{M}$ of H such that

$$K^i f \to A^{-1} f \quad \text{as} \quad i \to \infty,$$

for any element $f \in \overline{M}$.

proof

For simplicity we shall assume that K^0 is an identity operator.

By Theorem 3.1, K^i converge to an operator K on H. Operate A on the formula (3.15) from the right-hand side; then

$$AKAp_N{}^i = Ap_N{}^i. \tag{3.26}$$

Let M be a subset of H which consits of linear combinations of $Ap_N{}^i$, i=0,1,2,..., n.... Then the closure of M is clearly a subspace of H. The subspace is denoted by $\overline{M}$. We shall show that g^i and p^i, i=1,2,..., is an element of $\overline{M}$. From (3.24),

$$g^0 = \sum_{j=0}^{\infty} (g^0, p_N{}^j) Ap_N{}^j. \tag{3.27}$$

Hence,

$$g^i = \sum_{j=i}^{\infty} (g^0, p_N{}^j) Ap_N{}^j.$$

Substituting this into (3.25), we have

$$p^i = -K^i g^i$$

$$= -\sum_{j=i}^{+\infty} (g^0, p_N{}^j) Ap_n{}^j + \sum_{j=0}^{i-1} \beta_j Ap_N{}^j. \qquad (3.28)$$

This expression of p^i means that p^i is an element of $\bar{M}$. By Lemma 3.2 an element $f \in \bar{M}$ has the expression

$$f = \sum_{i=0}^{\infty} (f, p_N{}^i) Ap_N{}^i. \qquad (3.29)$$

From (3.29) we see that if an element $f \in \bar{M}$ is orthogonal to every $p_N{}^i$, $i=0,\ldots,$ then $f=0$. Hence $p_N{}^i$, $i=0,1,\ldots,$ is a complete system on $\bar{M}$. Then,

$$f = \sum_{i=0}^{\infty} (f, Ap_N{}^i) p_N{}^i \qquad (3.30)$$

by Lemma 3.3. Substitute (3.29) and (3.30) into (3.15) and (3.26); then

$$KAf = f, \quad f \in \bar{M}, \qquad (3.31)$$

$$AKf = f \quad f \in \bar{M}. \qquad (3.32)$$

Let $K_{\bar{M}}$ and $A_{\bar{M}}$ be operators on $\bar{M}$ such that

$$K_{\bar{M}} f = Kf \ ,$$

$$A_{\bar{M}} f = Af \quad \text{for} \quad f \in \bar{M}.$$

Then (3.31) and (3.32) show that

$$K_{\bar{M}} = A_{\bar{M}}{}^{-1}. \qquad (3.33)$$

In other words,

$$\lim_{i\to\infty} K^i f = A^{-1} f \quad \text{for} \quad f \in \bar{M}. \tag{3.34}$$

This completes the proof.

Let V be a sphere on $\bar{M}$, i.e.,

$$V = \{f: f\in\bar{M},\ |f|\leq 1\}.$$

If the convergence of (3.34) is uniform on V, the direction of the search in this method converges to that of Newton's method;

$$\frac{p^i}{|g^i|} = -K^i \frac{g^i}{|g^i|} \to \frac{-A^{-1}g^i}{|g^i|}.$$

3.5 Conjugate Gradient Method

Fletcher and Reeves's conjugate gradient method (Ref. 25) can be extended formally to Hilbert space H. The procedures of the method are the following;

$$\begin{aligned} p^0 &= -g^0 \\ p^i &= -g^i + \beta^{i-1} p^{i\ 1}, \end{aligned} \tag{3.35}$$

where

$$\beta^{i-1} = \frac{|g^i|^2}{|g^{i-1}|^2}. \tag{3.36}$$

By the formulas (3.35) and (3.36)

$$p^i = -|g^i|^2 \sum_{k=0}^{i} \frac{g^k}{|g^k|^2} \quad . \tag{3.37}$$

On the other hand, it is known that search directions of Davidon's method for quadratic functionals are expressed as follows (Ref. 29);

$$p^i = -|K^i g^i|^2 K^0 \sum_{k=0}^{i} \frac{g^k}{(g^k, K^0 g^k)} \quad . \tag{3.38}$$

If $K^0=I$ in Davidon's method, the search directions p^i in both methods are identical. So that, both algorithms generate a unique sequence $(u^0, u^1, \ldots)$ if an identical initial estimate u^0 is given. Now, we generalize the algorithm defined by (3.35) and (3.36);

$$\begin{aligned} p^0 &= -K^0 g^0 \\ p^{i+1} &= -K^0 g^{i+1} + \beta^i p^i \end{aligned} \tag{3.39}$$

$$\beta^i = \frac{|\sqrt{K^0} g^{i+1}|^2}{|\sqrt{K^0} g^i|} \quad . \tag{3.40}$$

Then,

$$\begin{aligned} p^{i+1} &= -K^0 g^{i+1} + \beta^i p^i \\ &= \{-K^0 g^{i+1} + \frac{(K^0 g^{i+1}, g^{i+1})}{(K^0 g^i, g^i)}\} \\ &\quad \times \{-K^0 g^{i+1} + \frac{(K^0 g^i, g^i)}{(K^0 g^{i-1}, g^{i-1})} p^{i-1}\} \end{aligned}$$

$$= -(K^0 g^{i+1},\ g^{i+1}) K^0 \sum_{k=0}^{i+1} \{\frac{g^k}{(K^0 g^k, g^k)}\} \ ,$$

that is,

$$p^i = -\sqrt{K^0}\ g^i|^2\ K^0 \sum_{k=0}^{i} \{\frac{g^k}{(K^0 g^k, g^k)}\} \ . \tag{3.41}$$

From (3.41) we can see that Davidon's method and (generalized) conjugate gradient method applied to quadratic functionals produce the same searching direction for the same initial estimate and the initial operator K^0. Thus, properties of Davidon's method presented in the preceeding section hold also for the conjugate gradient method defined by (3.39) and (3.40). For general nonquadratic functionals, the following algorithm which is a version of the conjugate gradient method is defined;

$$p^0 = -g^0$$

$$p^{i+1} = \gamma^{i+1}(-g^{i+1}+\beta^i p^i) \tag{3.42}$$

$$\beta^i = \frac{|g^{i+1}|^2}{|g^i|^2} \tag{3.43}$$

$$\gamma^i = \sqrt{\frac{1}{1+\beta^{i\ 1}}} \ . \tag{3.44}$$

For this algorithm if $|g^i| = |p^i|$,

$$|p^{i+1}|^2 = (\gamma^{i+1})^2\ (|g^{i+1}|^2 + \beta^{i2}|p^i|^2)$$

$$= (\gamma^{i+1})^2 |g^{i+1}|^2 (1+\beta^i)$$

$$= |g^{i+1}|^2,$$

$$\text{i.e.} \quad |p^{i+1}| = |g^{i+1}|, \tag{3.45}$$

and

$$(p^i, g^i) = -\gamma^i |g^i|^2. \tag{3.46}$$

Concerning the convergence of this method next results are obtained. Let's $S(u)$ be a functional bounded from below and define the domain D: $D \equiv \{u: S(u) \leq S(u^0)\}$. It is supposed that $S(u)$ have the first and second Fréshet derivatives denoted by $S'(u, h)$ and $S''(u, x, y) = (x, P(u)y)$.

Theorem 3.5

If D is bounded and convex and if there exist constants $m \geq 0$ and $M > 0$ such that

$$m|h|^2 \leq (h, P(u)h) \leq M|h|^2 \tag{3.47}$$

for any $h \in H$, $u \in D$. Then,

$$|g^i| \to 0 \quad \text{as} \quad i \to \infty \tag{3.48}$$

$$\lim_{i \to \infty} S(u^i) = \inf_{u \in D} S(u) \tag{3.49}$$

proof

$$S(u^k) - S(u^k + \alpha p^k)$$

$$= \alpha \gamma^k |g^k|^2 - \frac{1}{2}\alpha^2 (p^k, P(\xi^k) p^k)$$

$$\geq \alpha\gamma^k |g^k|^2 - \frac{1}{2}M\alpha^2 |p^k|^2.$$

where

$$\xi^k = u^k + \theta\alpha p^k, \qquad |\theta| < 1.$$

Let $\alpha_k^* = \frac{\gamma^k}{M}$, then

$$S(u^k) - S(u^k + \alpha^k p^k) \geq S(u^k) - S(u^k + \alpha_k^* p^k) \geq |g^k|^2 \left(\frac{(\gamma^k)^2}{2M}\right).$$

Suppose that there exists a constant N such that

$$(\gamma^k)^2 = \frac{1}{1+\beta^{k-1}} > N, \quad (k=0,1, \ldots).$$

Then $S(u^k) \to -\infty$ $(k \to +\infty)$ if $|g^k|$ does not converge to zero.

But this contradicts to the assumption on S(u). Hence we shall show that there exists a constant N, that is; $\beta^k < +\infty$. Since D is bounded $|\alpha^k p^k|$ is bounded uniformly with respect to k. So that, we may assume that $|\alpha^k|$ is also uniformly bounded. For, if not so $|p^k| = |g^k| \to 0$. On the other hand

$$(g^{k+1}, h) = (g^k, h) + \alpha^k (p^k, P^k(\eta^k)h)$$

where

$$\eta^k = u^k + \alpha^k \theta p^k, \quad (|\theta| \leq 1).$$

If $h = g^{k+1}/|g^{k+1}|$ in the above formula,

$$|g^{k+1}|=(g^k, \frac{g^{k+1}}{|g^{k+1}|})+\alpha^k(p^k, P(\eta^k)g^{k+1}/|g^{k+1}|)$$

$$\leq |g^k|+|\alpha^k||P(\eta^k)||p^k|.$$

Hence $\sqrt{\beta^k}\leq(1+|\alpha^k|M)$. Therefore $|\beta^k|$ is bounded uniformly with respect to k.

Next we shall show the second part of the theorem. By the assumption that $S(u)>-\infty$ and the above discussions $S(u^k)\geq S(u^{k+1})$. So that, there exists a constant L such that $\lim_{k\to\infty} S(u^k)=L$. Suppose that $L\neq \inf_{u\in D} S(u)$ Then there exist a point $z\in D$ such that $S(z)<L$. And, $0>S(z)-S(u^k)\geq(g^k, z-u^k)$. Since $\{u^k;\ k=0,1,\ldots\}$ is bounded and $|g^k|\to 0$, $0\geq S(z)-L\geq 0$. This is a contradiction. Hence $L=\inf_{u\in D} S(u)$

Theorem 3.6

Assume that the conditions in Theorem 3.5 are satisfied.

(1) If $m>0$ then there exists $z\in D$ such that $u^k\to z$ as $k\to\infty$, and z is uniquely determined.

(2) If $m\geq 0$ there is a subsequence $\{\bar{u}_0, \bar{u}_1,\ldots\}$ of the sequence $\{u_0, u_1,\ldots\}$ such that $\bar{u}_k$ weakly converge to a point $z\in D$ such that $S(z)=\inf_{u\in D} S(u)$ and $g(z)=0$.

proof

(1) At first we shall show that $\{u^k\}$ is a Cachey sequence. If it is not, for given $\varepsilon>0$ there exists a constant K such that

$$|u^s - u^k| \geq \varepsilon \quad \text{for} \quad s>k>K.$$

Then

$$0>S(u^s) - S(u^k)$$

$$= (g^k, u^s - u^k) + \frac{1}{2}(u^k - u^s, P(\xi)(u^k - u^s)),$$

$$\xi \in D.$$

Since D is bounded there is a constant δ such that

$$|u^s - u^k| \leq \delta, \quad (s>k>K)$$

If we take K so that $|g^k| \leq m\varepsilon^2/4\delta$,

$$(g^k, u^s - u^k) \geq -|g^k||u^s - u^k|$$

$$\geq -|g^k|\delta \geq -\frac{1}{4} m\varepsilon^2.$$

Hence, from the above relations,

$$S(u^s) - S(u^k) \geq \frac{m\varepsilon^2}{4} .$$

This contradicts to the convergence of $S(u^k)$. Therefore $\{u^k\}$ is a Cauchey sequence. So that there is a element $z \in D$ to which $\{u^k\}$ converges. Moreover

$$S(u) - S(z) \geq \frac{1}{2}|u-z|^2 m ,$$

for $u \in D$, so that z is uniquely determined.

(2) Since D is weakly compact by assumptions, there is a subsequence $\{\bar{u}^k\}$ of the sequence $\{u^k\}$ and $\{\bar{u}^k\}$ converges weakly to a element $z \in D$. Since $S''(u,h,h) \geq 0$, S(u) is

weakly lower semi-continuous (Ref. 49). Therefore

$$S(z) \leq \lim_{k\to\infty} S(\bar{u}^k) = L.$$

On the other hand $S(z) \geq L$. Hence $S(z)=L$ and clearly $g(z) \neq 0$.

At the end of this chapter, we shall give some comments on recent results concerning the convergence of Fletcher-Reeves's conjugate-gradient method and Davidon's method in R^n (Ref. 51). A convergence proof of the algorithm, which is a modification of Fletcher and Reeves's conjugate-gradient method, is presented by E. Polak. That is, if the objective function in R^n is strictly convex and twice continuously differentiable, then the sequence generated by the algorithm converges to the extremum point. The same convergence properties are proved by M. J. D. Powell for Davidon's method under the same assumptions. Also the rates of convergence of both methods are obtained; the infinite sequences constructed by these methods converge "superlinearly" when applied to strictly convex functions in R^n under some additional conditions. However, so far convergence of these methods is not known when applied to the minimization of a non-convex function in R^n. And also nothing is known concerning to the convergence of these method applied to non-quadratic functionals in function spaces.

3.6 Conclusions

Minimization problems in Hilbert space are discussed. Davidon's method and the conjugate gradient method in finite dimensional spaces are extended to the problems in Hilbert space. The stability and the convergency of the methods are studied for the case of quadratic functionals. And it is proved that the methods are stable from any initial approximation and that the sequences of the points of iterations converge to the true solution of the problem. It is also shown that the directions of the search converge to that of the Newton's method. Hence the schemes have the analogous property with Newton's method in the neighborhood of the extremal point. Stability of Davidon's method is also assured for nonquadratic problems, and so this method can be applied to such problems. A variant of Fletcher-Reeves's conjugate gradient method is presented. And the convergency for convex nonquadratic functionals is proved.

From the discussions for quadratic problems, we can say that Davidon's method has stability properties like those of the steepest descent method and that the convergence property in the vicinity of the extremum point is expected to be similar to that of the Newton's method.

CHAPTER IV

APPLICATIONS TO OPTIMAL CONTROL PROBLEMS

4.1. Unconstrained Continuous Optimal Control Problems

A control system is described by a system of ordinary differential equations;

$$\dot{x} = f(x, t, u) \tag{4.1}$$

where $x \in R^n$ is a state vector $u \in R^r$ is a control vector. Then, the problem is to find a control function $u=u^*(t)$ which minimizes the value of the function;

$$P(x(t_f)), \tag{4.2}$$

subject to (4.1) with an initial condition $x(t_0)=x^0$.
The following conditions are assumed.

(i) $f(x,u,t)$ and $P(x)$ have continuous partial derivatives of at least third order in all variables.

(ii) Optimal control $u=u^*(t)$ exists and is unique.

(iii) There are no constraints for x and u.

Let H be a space of r-dimensional control vector functions such that

$$\int_{t_0}^{t_f} \sum_{i=1}^{r} u_i^2(\tau)d\tau < \infty \ . \tag{4.3}$$

Then the space H is a Hilbert space with innerproduct

$$(u,\ v) = \int_{t_0}^{t_f} \sum_{1}^{r} u_i(\tau)v_i(\tau)d\tau. \tag{4.4}$$

Now, introduce an auxililary vector $\psi=(\psi_1,\ldots,\ \psi_n)$ and a Hamiltonian $\mathcal{H}(x,\ \psi,\ u,\ t)$ defined as follows;

$$\mathcal{H}(x,\ \psi,\ u,\ t) = \sum_{1}^{n} \psi_i f_i(x,\ u,\ t) \tag{4.5}$$

$$\dot{\psi}_i = -\sum_{j=1}^{n} \frac{\partial f_j(x,u,t)}{\partial x_i}\psi_j \qquad (i=1,2,\ldots,\ n) \tag{4.6}$$

$$\psi_i(t_f) = \frac{\partial P(x(t_f))}{\partial x_i(t_f)} \qquad (i=1,2,\ldots,\ n). \tag{4.7}$$

The equations (4.1) and (4.6) can be written with Hamiltonian in canonical form.

$$\dot{x} = \frac{\partial \mathcal{H}(x,\psi,u,t)}{\partial \psi}\ , \quad x(t_0)=x^0\ , \tag{4.8}$$

$$\dot{\psi} = -\frac{\partial \mathcal{H}(x,\psi,u,t)}{\partial x}\ , \quad \psi(t_f)=\frac{\partial P}{\partial x}\ . \tag{4.9}$$

Let $x(t)$ and $\psi(t)$ be a solution of the equations (4.8) and (4.9) corresponding to a certain control $u(t)$. Since terminal state $x(t_f)$ is determined by the given control $u(t)$, the performance index $P(x_f)$ is a functional of $u(\cdot)\in H$. We denote this functional by $J(u)$

$$J(u) = P(x(t_f)) \tag{4.10}$$

Therefore the optimal control problem is reduced to the minimization problem of a functional. Hence, the methods of the preceding sections are applicable. The gradient g(t) of the functional J(u) is determined by the definition;

$$\lim_{\varepsilon \to 0} \frac{J(u+h)-J(u)}{\varepsilon} = \int_{t_0}^{t_f} g^T(\tau)h(\tau)d\tau. \qquad (4.11)$$

Then (Ref. 9),

$$g(t) = \frac{\partial \mathcal{H}(x(t),\psi(t),u(t),t)}{\partial u}. \qquad (4.12)$$

4.2. Unconstrained Discrete Optimal Control Problems

A control system is described by a system of difference equations

$$x(i+1)-x(i) = f_i(x(i),u(i)), \qquad (4.13)$$

where $x(i) \in R^n$, $u(i) \in R^r$ and i=1,2, .., N. Then, the problem is to find a set of control (u(0), u(1),..., u(N-1)) which minimizes the function

$$P(x(N)) \qquad (4.14)$$

subject to (4.13) with an initial condition $x(0)=x^0$. Thus discrete optimal control problems can be viewed as nonlinear programming problems if we define a function J of N×r variables

such that

$$J(u(0),\ldots\ u(N-1) \equiv P(x(N)). \tag{4.15}$$

And the gradient of $J(u(0),\ldots, u(N-1))$, denoted by $g(z)$ with $z\equiv(u(0), .., u(N-1))$, can be computed by the following procedures (Ref. 52).

At first solve the following equations

$$\psi(i) - \psi(i+1)=\left(\frac{\partial f_i(x(i),u(i))}{\partial x(i)}\right)^T\psi(i+1), \tag{4.16}$$

$$(i=0,1,\ldots, N-1),$$

with

$$\psi(N) = -\left(\frac{\partial P(x(N))}{\partial x(N)}\right)^T. \tag{4.17}$$

Then

$$\left(\frac{\partial J(Z)}{\partial u(i)}\right)^T = -\left(\frac{\partial f_i(x(i),u(i))}{\partial u(i)}\right)^T \psi(i+1), \tag{4.18}$$

$$(i=0,1, .., N-1).$$

4.3. Constrained Optimal Control Problems

Consider the following constrained problem. Minimize

$$P(x_f, t_f) \tag{4.19}$$

subject to

$$\dot{x}=f(x, u, t) \tag{4.20}$$

$$x(t_0)=x^0 \tag{4.21}$$

$$h(x_f, t_f)=0 \tag{4.22}$$

$$g(x, u, t)\geq 0 , \tag{4.23}$$

where x is an n-dimensional vector, $x_f=x(t_f)$, u is an r-dimensional vector, h is an m-vector, and g is a s-vector of functions, where m<n. The final time t_f is determined by the condition (4.22). This constrained problem is transformed to problems without inequality constraints by adding a penalty function to the initial performance function (4.19). The problem becomes to minimize functional;

$$J(u,r) = P(x_f,t_f)+\gamma \sum_{i=1}^{s} \int_{t_0}^{t_f} \frac{dt}{g_i(x,u,t)}, \qquad (4.24)$$

where γ is a positive scalar. The computing procedures become as following. Choose $\gamma_1>0$ and initial control u^0 such that $g_i(x(t), u^0(t), t)>0$, $t_0\leq t\leq t_f$, i=1,..., s, and $h(x(t), u^0(t))\neq 0$, $t_0\leq t\leq t_f$, where x(t) is a solution of (4.26) corresponding to $u^0(t)$. And consider the problem of minimizing $J(u, \gamma_1)$, starting from u^0, subject to the equation (4.20) and the terminal constraint (4.22). The same procedures are repeated with $\gamma_1>\gamma_2>..>\gamma_k>0$. Several theoretical results are known concerning to this penalty methods (Ref. 21, 53, and 54)

For example, under some assumptions (Ref. 21),

(1) $$\lim_{\gamma_k\to 0} (\min_{u\in S} J(u, \gamma_k)) = \inf_{u\in S} P(x_f, t_f), \qquad (4.25)$$

(2) $$\lim_{\gamma_k\to 0} P(\bar{u}^k, t_f) = \inf_{u\in S} P(x_f, t_f), \quad \text{and} \qquad (4.26)$$

$$(3)\quad \lim_{\gamma_k \to 0} \gamma_k \sum_{i=0}^{s} \int_{t_0}^{t_f(\gamma_k)} \frac{dt}{g_i(\bar{u}^k, x_k, t)} = 0, \qquad (4.27)$$

were S is a set of control which satisfies the given constraints and $\bar{u}^k$ is a solution of the transformed problem with $\gamma=\gamma_k$.

4.4 Examples

Example 1.

Consider a control system

$$\dot{x}_1 = x_2,$$

$$\dot{x}_2 = -x_1 + (1-x_1^2)x_2 + u, \quad x_1(0) = x_{10}, \quad x_2(0) = x_{20},$$

with a performance index

$$J = \int_0^5 (x_1^2 + x_2^2 + u^2)dt.$$

The control time is fixed as $t_0=0$, $t_f=5$. We introduce the third coordinate x_3 such that

$$\dot{x}_3 = x_1^2 + x_2^2 + u^2, \quad x_3(0) = 0.$$

The canonical equation then becomes:

$$\begin{aligned} \dot{x}_1 &= x_2, & x_1(0) &= x_{10}, \\ \dot{x}_2 &= -x_1 + (1-x_1^2)x_2 + u, & x_2(0) &= x_{20}, \\ \dot{x}_3 &= x_1^2 + x_2^2 + u^2, & x_3(0) &= 0, \end{aligned}$$

$$\dot{\psi}_1 = (1 + 2x_1x_2)\psi_2 - 2x_1\psi_3, \qquad \psi_1(5) = 0,$$

$$\dot{\psi}_2 = -\psi_1-(1-x_1^2)\psi_2 - 2x_2\psi_3, \qquad \psi_2(5) = 0,$$

$$\dot{\psi}_3 = 0, \qquad \psi_3(5) = 1.$$

The computed results for $x_{10}=3.0$, $x_{20}=0.0$ are shown in Fig. 1 and Fig. 2. The results obtained by the steepest descent method and Fletcher and Reeves's conjugate gradient method are also shown.

Example 2.

$$\dot{x}_1 = x_2,$$

$$\dot{x}_2 = -0.2x_2 + 2.0x_3 - 0.2x_2x_3^2,$$

$$\dot{x}_3 = -5x_2 + u,$$

$$J = \int_0^5 (x_1^2 + x_2^2 + x_3^2 + u^2)dt.$$

The numerical results are shown in Fig. 3 and Fig. 4 with $x_{10}=0.25$, $x_{20}=0.25$, $x_{30}=0.1$.

From these examples we can say that Davidon's method proposed here is applicable to nonlinear control problems and the rapid convergence is assured for these examples. These results also show that the conjugate gradient method is also a very useful scheme.

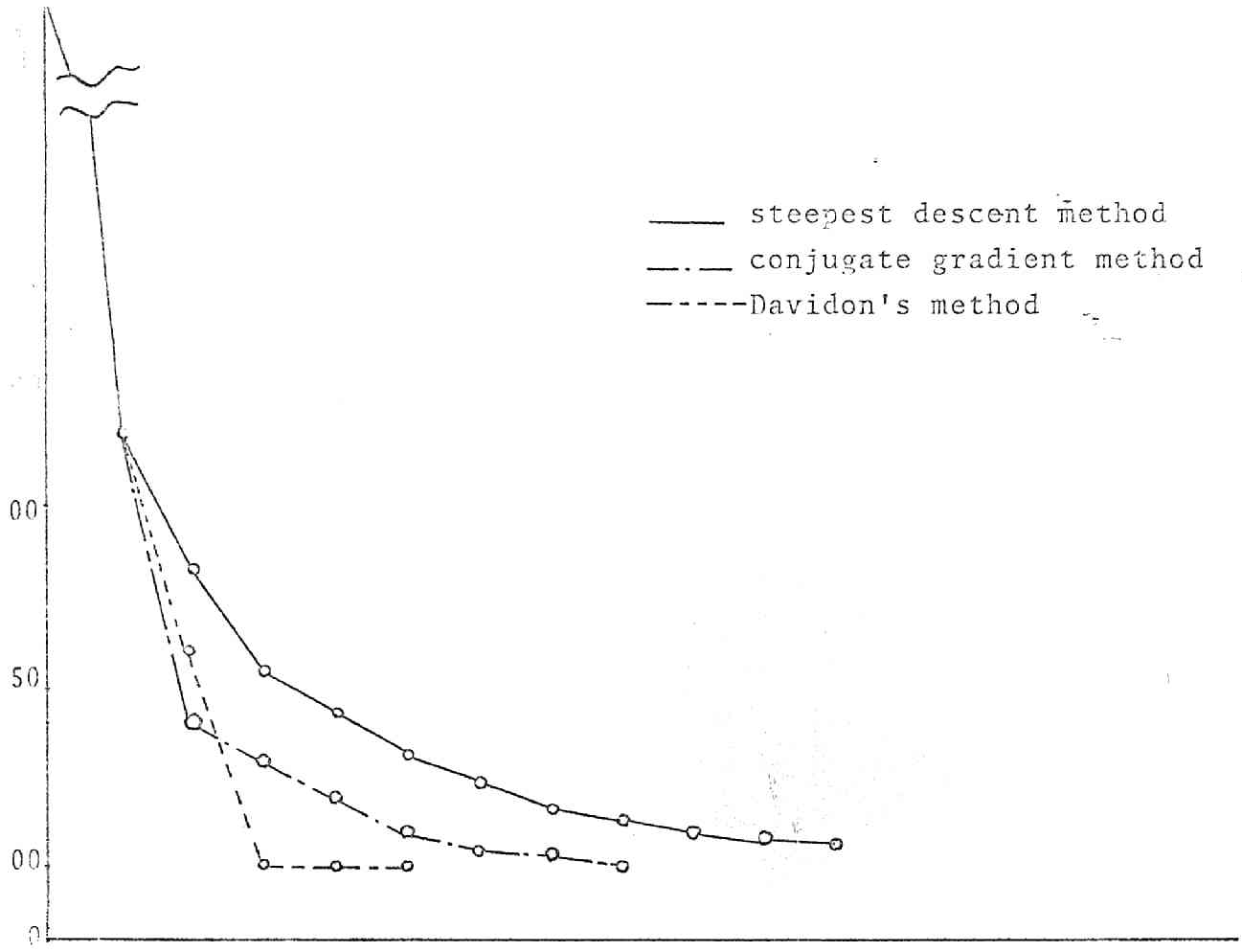

Fig.1 (values of performance index)

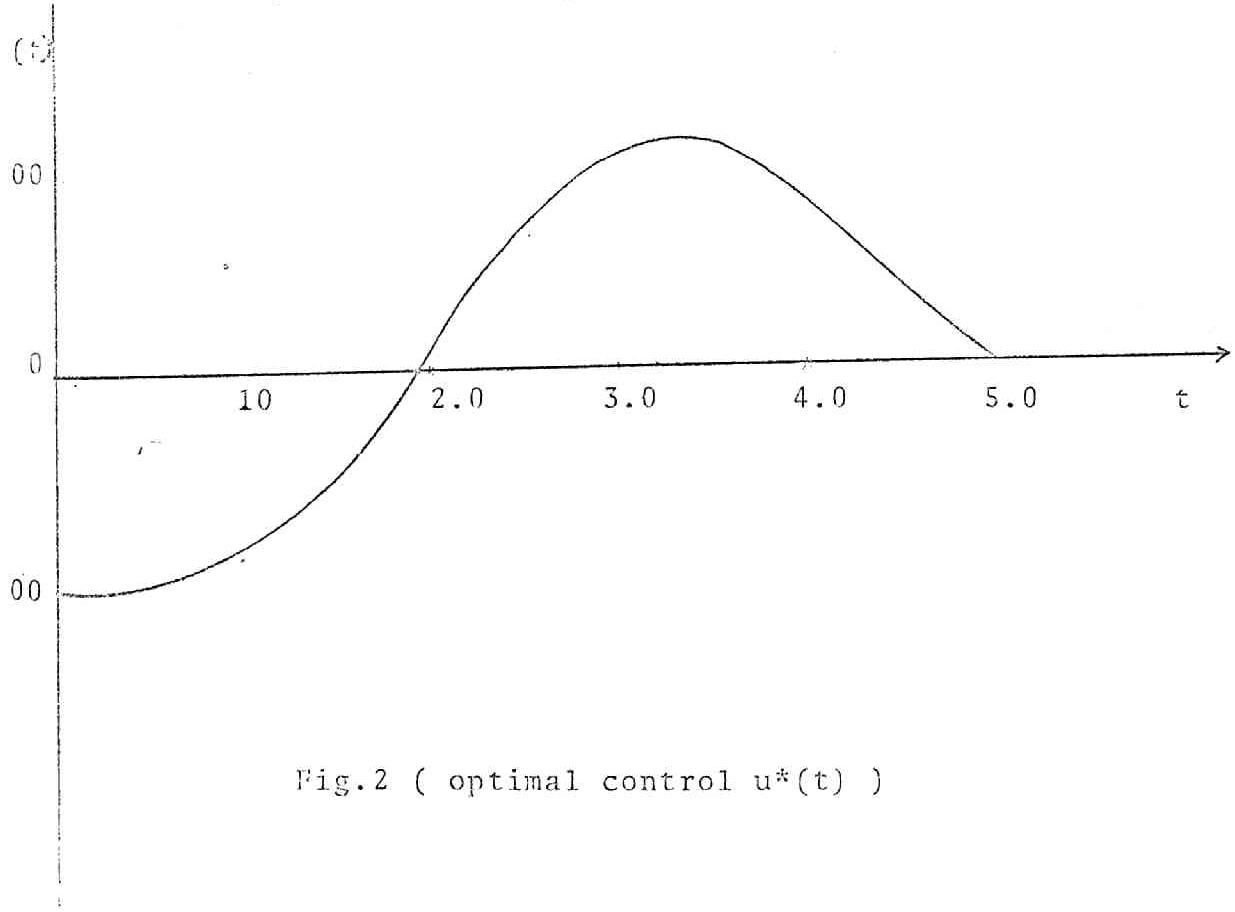

Fig.2 (optimal control u*(t))

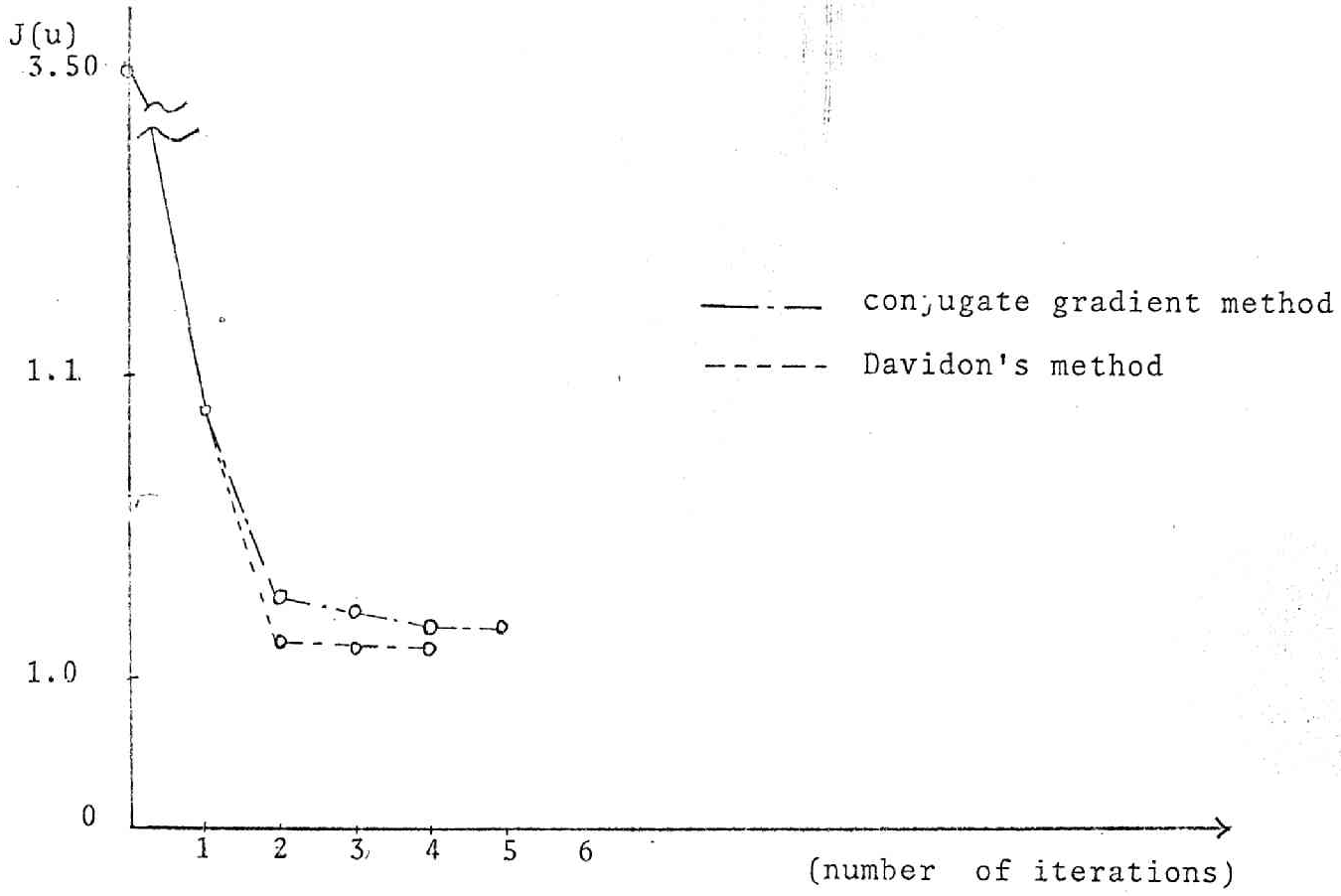

Fig.3 (values of performance index)

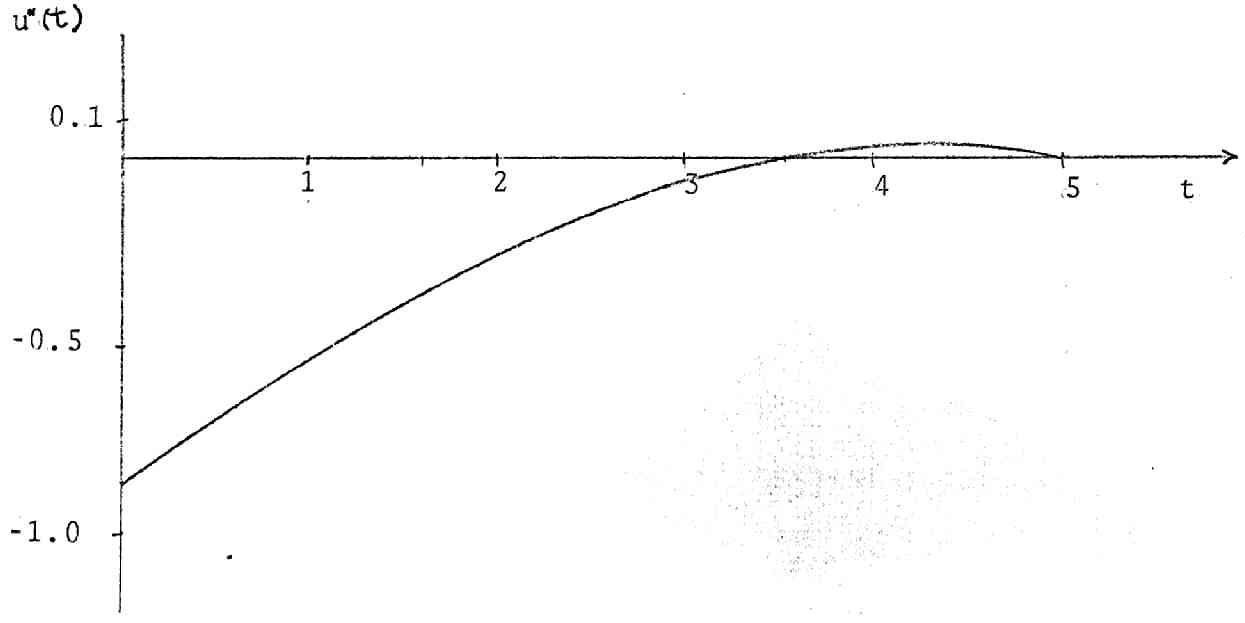

Fig.4 (optimal control u*(t))

4.5 Conclusions

Optimal control problems are reduced to minimization problems of functionals in finite or infinite dimensional spaces. Therefore numerical methods developed in the preceeding chapter can be applied to the problems. Two numerical examples are shown. These examples show the superiority of Davidon's method, compared with the steepest descent method or the conjugate gradient method. But the convergency of the method for general nonquadratic functional is not proved. The disadvantage of Davidon's method is that the information to be stored in the computer increases with the number of iterations. So, if convergence is slow, computing will be difficult. The conjugate gradient method is inferior to Davidon's method with respect to the rate of convergence, but it has great advantages that the algorithm is very simple and that a small amount of the storage of the information is required. Continuous control problems are approximated by discrete problems in the computations of the examples, but the computations of the transformed problems by Davidon's method will be difficult for large control systems by the limitation of the capacity of the storage. Therefore other approaches, such as functional approximations of controls, are to be developed.

There is a theoretical problem, which is not discussed in this paper, in numerically solving the continuous optimal

control problems. That is, the original continuous problem is replaced by a discrete problem and there arises the question of the convergence of the solution of the "difference" problem to the solution of the "differential" problem. If the answer of the above question is affirmative the original problem is said "well-posed". Some resonable conditions for well-posedness are obtained (Ref. 53, 54). But more researches for this problem are required.

REFERENCES

(1) L. S. Pontrayagin, V. G. Boltyanskii, R. V. Gamkrelize, and E. F. Mischchenko, "The Mathematical Theory of Optimal Processes". Wilely, New York, 1962.

(2) R. Bellman, "Dynamic Programming", Princeton Univ. Press, Princeton, New Jersey, 1957.

(3) P. Kennth and R. McGill, Two Point Boundary-Value-Problem Technique, in "Advances in Control Systems", Vol. 3, (C.T. Leondes ed.), Academic Press, New York, 1966.

(4) D. K. Scharmack, An Initial Value Method for Trajectory Optimization Problems, in "Advances in Control Systems", Vol. 5, (C.T. Leondes ed.), Academic Press, New York, 1967.

(5) D. Issacs, Algorithms for Sequential Optimization of Control Systems, in "Advances in Control Systems", Vol. 4, (C.T. Leondes ed.), Academic Press, New York, 1967.

(6) A. E. Bryson and W. F. Denham, A Steepest Ascent Method for Solving Optimal Programming Problems, J. Appl. Mech., Vol. 29, No. 2, pp. 247-257, 1962.

(7) H. J. Kelley, Method of Gradients, in "Optimization Techniques" (G. Leitman, ed.), Academic Press, New York, 1962.

(8) J. V. Breakwell, J. L. Speyer and A. E. Bryson, Optimization and Control of Nonlinear Systems Using the Second Variation, J. SIAM Control, Vol. 1, No. 2, pp. 193-223, 1963.

(9) R. E. Kopp and R. McGill, Several Trajectory Optimization Techniques, in "Computing Methods in Optimization Problems" (A. V. Balakrishnam, ed.), Academic Press, New York, 1964.

(10) C. W. Merrian, "Optimization Theory and the Design of Feedback Control", McGraw Hill, 1964.

(11) S. K. Mitter, Sucessive Approximation Method for the Solution of Optimal Control Problems, Automatica, Vol. 3, pp. 135-149, 1966.

(12) L. A. Zadeh, On Optimal Control and Linear Programming, Correspondence to IRE Trans. on AC, Vol. AC-7, No. 4, pp. 45-46, 1962.

(13) G. B. Danzing, Linear Control Processes and Mathematical Programming, J. SIAM Control, Vol. 4, No. 1, 1966.

(14) J. B. Rosen, The Gradient Projection Method for Nonlinear Programming, SIAM J. on Appl. Math., Vol. 8, No. 1, pp. 181-217, 1960.

(15) A. V. Fiacco and G. P. McCormic, "Nonlinear Programming; Sequential Unconstrained Minimization Techniques", John Wiley and Sons, Inc., 1968.

(16) J. B. Rosen, Optimal Control and Convex Programming, in "Nonlinear Programming" (J. Abadie, ed.), North-Holland, 1967.

(17) D. Tabak, Application of Mathematical Programming in the Design of Optimal Control Systems, Int. J. on Control, Vol. 10, No. 5, pp. 545-552, 1969.

(18) D. Tabak, Optimal Control of Nonlinear Discrete Time System by Mathematical Programming, J. of the Franklin Institute, Vol. 289, No. 2, pp. 111-119, 1970.

(19) E. S. Levitin and B. T. Polyak, Constrained Minimization Methods, Zh. Vȳchisl. Mat. & Mat. Fiz., Vol. 6, No. 5, pp. 787-823, 1966.

(20) J. Gera, Branched Trajectory Optimization by the Projected Gradient Technique, AIAA Journal Vol. 8, No. 6, pp. 1121-1126, 1970.

(21) L. S. Lasdon, A. D. Waren, and R. K. Rice, An Interror Penalty Method for Inequality Constrained Optimal Control Problems, IEEE Transaction on AC., Vol. AC-12, No. 4, pp. 388-395, 1967.

(22) A. V. Balakrishnan, On a New Computing Method in Optimal Control, SIAM J. on Control, Vol. 6, No. 2, pp. 149-173, 1968.

(23) A. P. Jones and G. P. McCormik, A Generalization of the Method of Balakrishnan: Inequality Constraints and Initial Conditions, SIAM J. Control Vol. 8, No. 2, pp. 218-224, 1970.

(24) R. G. Brusch and R. H. Schappele, Solution of Highly Constrained Optimal Control Problems Using Nonlinear Programming, AIAA paper No. 70-964, Presented at AIAA Guidance, Control and Flight Mechanics Conference Santa Barbara, 1970.

(25) R. Fletcher and G. M. Reeves, Functional Minimization by Conjugate Gradients, Compt. J. Vol. 7, pp. 149-154, 1964.

(26) R. Fletcher and M. J. D. Powell, A Rapidly Convergent Descent Method for Minimization, Compt. J. Vol. 6, pp. 163-168, 1963.

(27) J. H. Westcott, Computational Methods of Optimization in Control, Survay Paper in 4-th IFAC Congress, 1969.

(28) L. S. Lasdon, S. K. Mitter and A. D. Waren, The Conjugate Gradient Method for Optimal Control Problems, IEEE Trans. on AC, Vol. AC-12, pp. 132-138, 1967.

(29) L. B. Horowitz and P. E. Sarachir, Davidon's Method in Hilbert Space, SIAM J. Appl. Math. Vol. 16, No. 4, pp. 677-694, 1968.

(30) J. F. Sinnot, Jr., Solution of Optimal Control Problems by the Method of Conjugate Gradients, Preprint of JACC, 1967.

(31) W. C. Davidon, Variable-Metric Method for Minimization, Argonne National Laboratory, Report No. ANL-5990, 1959.

(32) M. R. Hestens and E. Stiefel, Methods of Conjugate Gradients for Solving Linear Systems, Journal of Research of the National Bureau of Standards, Vol. 49, No. 6, pp. 409-436, 1952.

(33) C. G. Broyden, Quasi-Newton Methods and Their Application to Function Minimization, Mathematics of Computation, Vol. 21, No. 99, pp. 368-381, 1967.

(34) G. P. McCormik and J. D. Pearson, Variable Metric Methods and Unconstrained Optimization, in "Optimization" (R. Fletcher, ed.), Academic Press, New York, 1969.

(35) D. Goldfarb, Sufficient Conditions for the Convergence of Variable-Metric Algorithm, in"Optimization" (R. Fletcher, ed.), Academic Press, New York, 1969.

(36) J. D. Pearson, On Variable-Metric Methods of Minimization, Computer Journal, Vol. 12, No. 2, pp. 171-178, 1969

(37) J Greenstadt, Variations on the Variable-Metric Method, Mathematics of Computation, Vol. 24, No. 109, pp. 1-21, 1970.

(38) D. GoldFarb, A Family of Variable-Metric Methods Derived by Variational Means, Mathematics of Computation, Vol. 24, No. 109, pp. 23-26, 1970.

(39) H. Y. Huang, Unified Approach to Quadratically Convergent Algorithms for Function Minimization, Journal of Optimization Theory and Applications, Vol. 5, No. 6, pp. 405-423, 1970.

(40) H. Y. Huang and A. V. Levy, Numerical Experiments on Quadratically Convergent Algorithms for Function Minimization, Journal of Optimization Theory and Applications, Vol. 6, No. 3, pp. 269-282, 1970.

(41) R. P. Tewarson and S. Brook, On the Use of Generalized Inverse in Function Minimization, Computing, Vol. 6, pp. 241-248, 1970.

(42) B. V. Sha, R. T. Bueher and O. Kempthorne, Some Algorithms for Minimizing a Function of Several Variables, SIAM J. on Appl. Math. Vol. 12, No. 1, pp. 74-92, 1962.

(43) M. J. D. Powell, An Iterative Method for Finding Stationary Values of a Function of Several Variables, Compt. J. Vol. 5, No. 2, pp. 147-151, 1962.

(44) R. Fletcher, A Review of Methods for Unconstrained Optimization, 'Optimization" (R. Fletcher, ed.), Academic Press, New York, 1969.

(45) R. Penrose, A Generalized Inverse for Matrices, Proceeding of the Cambridge Philosophical Society, Vol. 51, Part 3, pp. 406-419, 1954.

(46) B. A. Murtagh and R. W. H. Sargent, A Constrained Minimization Methods with Quadratic Convergence, "Optimization" (R. Fletcher, ed.), Academic Press, New York, 1969.

(47) J. Kowalik and M. R. Osborn, "Method for Unconstrained Optimization Problems" American Elsevier Published Company Inc., New York, 1968.

(48) B. Takamatsu, S. Sayama and K. Oh-i, Some Consideration on One Dimensional Search in Gradient Method, J. of the Japan Association of Automatic Control Engineers, (in Japanese), Vol. 13, No. 9, pp. 24-32, 1969.

(49) M. M. Vainberg, "Variational Methods for the Study of Nonlinear Operators, Holden-Day, Inc., San Fransisco, 1964.

(50) A. V Balakrishnan, A General Theory of Nonlinear Estimation Problems in Control Systems, Journal of Mathematical Analysis and Applications Vol. 8, pp. 4-30, 1964.

(51) E. Polak, Computational Methods in Optimization; a Unified Approach, Academic Press, New York and London, 1971.

(52) B. M. Budak, E. M. Berkovich and E. N. Solov'eva, Difference Approximations in Optimal Control Problems, SIAM J. Control, Vol. 7, No. 1, pp. 18-30, 1969.

(53) J Cullum, Discrete Approximations to Continuous Optimal Control Problems, SIAM J. Control, Vol. 7, No. 1, pp. 32-48, 1969.